CONTINUOUS STRATEGY DEVELOPMENT FOR EFFECTS-BASED OPERATIONS

BAE Systems Advanced Information Technologies, Inc.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2006-29 has been reviewed and is approved for publication

APPROVED: /s/

        JOSEPH A. CAROLI
        Project Engineer

FOR THE DIRECTOR: /s/

        JAMES W. CUSACK, Chief
        Information Systems Division
        Information Directorate

REPORT DOCUMENTATION PAGE		*Form Approved* *OMB No. 074-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE FEBRUARY 2006	3. REPORT TYPE AND DATES COVERED Final Feb 01 – Nov 05

4. TITLE AND SUBTITLE
CONTINUOUS STRATEGY DEVELOPMENT FOR EFFECTS-BASED OPERATIONS

6. AUTHOR(S)
Nicholas J. Pioch

5. FUNDING NUMBERS
C   - F30602-01-C-0047
PE - 63789F
PR - EBO0
TA - 00
WU - 02

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
BAE Systems
6 New England Executive Park Drive
Burlington Massachusetts 01803

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory/IFSA
525 Brooks Road
Rome New York 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2006-29

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Joseph A. Caroli/IFSA/(315) 330-4205/ Joseph.Caroli@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*
The primary objective of this program was to develop the Strategy Development Tool (SDT), a system to aid the Strategy Division of an Air Operations Center (AOC) in authoring robust effects-based plans and assessing their impact on enemy systems. To this end, we conducted research and development in five major stages. First, we developed an EBO plan representation and plan authoring tool for decomposing strategic objectives into desired effects, causal linkages, and tasks. Second, we integrated this plan editor with AFRL's Causal Analysis Tool (CAT) for assessment of course of action probability of success over time. Third, we developed an interface to an operational-level target system analysis and option generation capability known as Endstate. Fourth, we developed a strategic-level Center of Gravity (COG) Articulator tool for building lightweight causal models of enemy systems, along with a methodology by which blue interventions against these systems could be analyzed in the CAT and causal chains of effects resulting from interventions could be automatically imported into the SDT plan editor. Finally, we hardened the existing SDT editors, developed additional capabilities for collaborative planning and assessment, and developed new external interfaces for Joint Expeditionary Force Experiment (JEFX) 2004 to support and AFRL-led EBO initiative. This culminated in the Government's decision to transition portions of SDT into Information Warfare Planning Capability (IWPC) to enhance its capabilities for effects-based planning and real-time collaboration.

14. SUBJECT TERMS
Effects-Based Operations, Strategy Development, Collaborative Planning, Course of Action Authoring, Adversary Modeling, Dynamic Bayesian Networks, Joint Expeditionary Force Experiment, Target System Analysis

15. NUMBER OF PAGES
200

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SECTION 1

SUMMARY

This technical report describes work performed on the Continuous Strategy Development for Effects-Based Operations (contract number F30602-01-C-0047) project conducted for the Air Force Research Lab (AFRL)-Information Directorate in Rome, NY as part of the Effects-based Operations Advanced Technology Demonstration (EBO ATD). The primary objective of this program was to develop the Strategy Development Tool (SDT), a system to aid the Strategy Division of an Air Operations Center (AOC) in authoring robust effects-based plans and assessing their impact on enemy systems using a variety of strategic- and operational-level analysis tools. The effort culminated in participation in an initiative for Joint Expeditionary Forces Experiment (JEFX) 2004 focused on Predictive Battlespace Awareness (PBA) and EBO, which led to the Government's decision to transition portions of SDT into an existing AOC system of record, Information Warfare Planning Capability, in order to expand its support for EBO and real-time collaborative planning.

The first year of the program was focused primarily on establishing a suitable plan representation for effects-based air campaign plans, collaborating with consultant Dr. Maris (Buster) McCrabb and associate contractor ISX Corporation, and on developing a prototype Plan Editor component for SDT. The initial prototype leveraged a novel restricted natural language user interface tool developed under DARPA's Active Templates program for automatically guiding course of action (COA) statements. Upon completion of the prototype, we also developed an interface with AFRL's Causal Analysis Tool (CAT) to export the SDT-developed COA in the form of a dynamic Bayesian network with delays, persistences, and conditional causal probabilities. The CAT tool could then be used to predict the probability of the COA's success and its constituent plan elements (objectives, effects, tasks) over time. The SDT team also participated in AFRL's annual Cross Thrust demonstration, showcasing SDT, CAT, and a number of other related AFRL tools within an integrated "Operation Deny Force" scenario designed by Buster McCrabb. The Cross Thrust demonstration was performed on multiple occasions for the Air Force Scientific Advisory Board both at AFRL and during a technology symposium at Langley AFB.

As we continued to refine the Plan Editor in Year 2, we focused attention on adversary modeling and analysis by defining a new methodology for intervention analysis in CAT and integrating SDT with the operational-level target system analysis and option generation capabilities of Endstate, another EBO ATD effort funded simultaneously with SDT. We extended the Operation Deny Force scenario to include an enemy Center of Gravity (COG) model authored in CAT describing the notional enemy's temporal causal relationships among its Weapons of Mass Destruction, Electric Power, and Petroleum Oil Lubricants COGs. This model was also used as the Endstate Leadership model to govern enemy decisions on workaround policies in the face of simulated target-induced outages from the SDT plan. Endstate's resulting "outage profiles" were passed to SDT and displayed to the user as geographical overlays and standalone charts. In a later scenario branch covering option generation, SDT was modified to send only targets associated with desired effects and Endstate returned a list of proposed strike targets that the strategy user could incorporate into the plan. Year 2 culminated in the AFC2ISRC EBO Workout, a one-week experiment held at the Langley AFB Transformation Center to gain further insight into the EBO planning methodology.

Based in part on observations at the EBO Workout and lessons learned from initial modeling activities in CAT, we focused in Year 3 on a new strategic-level adversary modeling tool, the COG Articulator. The goal of the COG Articulator was to provide a lightweight brainstorming environment for causal modeling of enemy systems of systems and "what if" activities such as blue vs. red wargaming. Because the COG Articulator was built on top of SDT's existing interface to CAT, COG models could be instantly analyzed in CAT to predict the probability over time of enemy nodes representing actions, goals, resources, capabilities, or beliefs, comparing results with vs. without blue interventions or using different combinations and timing of blue interventions. COG Articulator was later extended to allow users to highlight causal chains of red system nodes stemming from blue intervention nodes and import them into the SDT plan as a blue COA fragment, based on the most favorable combination of interventions revealed by a comparative analysis in CAT.

Beginning in late 2003, we focused our main development efforts on preparing the software for participation in AFRL's JEFX '04 PBA-EBO initiative. The Plan Editor was enhanced with redesigned editor screens with cleaner labeling and separation of information than was possible using the original prototype's restricted natural language templates. We developed a message-based collaboration capability that was later redesigned based on early JEFX feedback into a more sophisticated real-time collaborative plan authoring engine allowing teams of users to edit the same plan with automatic updates to ensure synchronized views. We extended SDT's built-in assessment capabilities to include a COA Comparison Matrix and plan element assessment tabs showing completion percentages that could be graphically displayed in SDT's Timeline Editor. Finally, we developed and integrated numerous external tools and interfaces, including an XML-based plan interchange with TBMCS 1.1.3's Strategy Management System. In 2004, we provided multi-person on-site support for JEFX spirals 1-4, as well as significant SDT enhancements between spirals based on end-user feedback.

Prior to JEFX '04, an early transition of EBO planning and visualization concepts from SDT took place via the Electronic Systems Center's (ESC's) Strategy Planning Tool (SPT), which was included in TBMCS 1.1.3. After successful completion of JEFX '04, the last year of the program focused almost solely on transition activities. SDT's primary transition path took shape during a series of post-JEFX Strategy Requirements Working Group meetings led by AFC2ISRC, namely to incorporate selected capabilities of SDT into the ESC's existing Information Warfare Planning Capability (IWPC), developed by General Dynamics. Primary capabilities of interest include SDT's EBO plan representation and its real-time collaborative planning engine.

SECTION 2

INTRODUCTION

Effects-based approaches to planning, executing, and assessing air operations begin with the recognition that opponents are intelligent, devious, and proactive. It follows that in planning to achieve a given effect on an opponent, it is crucial to model their likely response, taking into account their doctrine, historical patterns of behavior, and current capabilities. Under contract to AFRL, BAE Systems has built a Strategy Development Tool (SDT) using an effects-based ontology for plan authoring, in which a course of action (COA) is refined into strategic and operational objectives, desired effects, and tasks, informed by models of the expected enemy response at each level of abstraction. The SDT uses temporal causal models to represent causal relationships, delays, and persistences within and across enemy centers of gravity. These models may be analyzed in AFRL's Causal Analysis Tool (CAT) to predict the probability of COA success over time. Operational-level models of enemy target systems are included in the Endstate system, integrated with SDT to provide predictions of outage levels over time caused by blue target sets as well as option generation to suggest strike targets to achieve specified effects against indirect effect targets. SDT was the centerpiece of AFRL's JEFX '04 PBA/EBO initiative, providing real-time collaborative mission analysis and strategy planning for the Strategy Division players at the Nellis AFB Combined Air Operations Center (CAOC). After successful completion of JEFX '04, portions of SDT are being transitioned into ESC's IWPC, and BAE Systems continues to work to advance the state-of-the-art in research on EBO-enabled tools in several follow-on programs.

The effort began by focusing on transitioning technology developed at ALPHATECH, Inc. (now BAE Systems) under DARPA's Active Templates program for grammar-constrained natural language interfaces and knowledge-based plan representation for ground forces operations. Working with our subject matter expert, Dr. Buster McCrabb, and co-contractor ISX Corporation, we developed a new plan representation that decomposes COAs into phases, objectives, desired effects, causal linkages and tasks. We then proceeded with incremental development of SDT as an integrated suite of the following prototype tools, described in more detail in the next section:

- Plan Editor – used for top-down or bottom-up strategy authoring, with multiple views of the overall plan and detailed editors for each plan element type.

- Plan Manager – an "internal" component that manages SDT's two primary external interfaces to the strategic- and operational-level adversary analysis tools CAT and Endstate.

- COG Articulator – a lightweight whiteboard-style tool for rapid authoring of causal adversary COG models, linked to CAT for probabilistic analysis and to the Plan Editor to import causal intervention chains as blue plan fragments.

- Collaboration Server – Originally conceived as a suite of tools for delegating plan fragments, this capability evolved into an engine for multi-user real-time collaborative plan updates based on BAE Systems' previously developed Adaptive Planning Software for Special Operations.

- External Interfaces – In addition to the existing CAT and Endstate interfaces, to prepare for EBO ATD demonstrations and JEFX '04, SDT has been integrated with the following external systems:

  - Dynamic Course of Action Decision Tool (DCOAD) – a PBA-focused tool also developed by BAE Systems that may be launched from SDT's analysis toolbar to support viewing of raw intelligence data from AFRL's A2IPB and prediction of gaps in ISR or strike coverage due to Time Sensitive Targets.

  - Effects-based Operations Wargaming System (EBOWS) – loosely integrated with SDT to read its XML-based COA description and wargame it against a pre-specified enemy military configuration. Developed by L3 Communications.

  - Strategy Management System (SMS) – Developed by Accenture for inclusion in TMBCS 1.1.3 and JEFX '04, SDT is able to translate its own XML format into that of SMS and upload plans to the SMS web service for consumption by other strategy tools such as IWPC or Strategy Planning Tool (SPT) as well as downstream targeting tools such as Joint Targeting Toolbox (JTT).

  - Fusion for EBO (FEBO) – another in-house tool funded by AFRL-IFE which reads the Measures of Effectiveness and indicators from SDT's plan representation, decomposes them into specific ISR tasking and then fuses incoming ISR data to assess the status of tasks and effects.

Throughout the EBO ATD, other tools and AFRL-funded programs have been loosely integrated with SDT to test their system using realistic effects-based plans, such as Securboration's Scenario Generation tool.

The next section provides a detailed discussion of the methodology used in developing SDT and integrating it with each of the above tools. The following section discusses results from participation in demonstrations and experiments, including AFRL's Cross Thrust, AFC2ISRC's EBO Workout, and JEFX '04, and also discusses technology transition activities. The report closes with conclusions drawn from these results, and recommendations for future action based on the results of the research.

4

SECTION 3

METHODS, ASSUMPTIONS, AND PROCEDURES

3.1   EBO PLAN REPRESENTATION

The first task addressed under this research was the specification of an appropriate effects-based plan representation for air campaigns. For this task, BAE Systems (then known as ALPHATECH, Inc.) initially adapted concepts from ground forces planning doctrine from our involvement in DARPA's High-Performance Knowledge Bases and Active Templates programs. The task was significantly aided by collaboration with our consultant Buster McCrabb and associate EBO ATD contractor ISX Corporation, who had developed an ontology for EBO planning under a Phase I Small Business Innovative Research (SBIR) effort.

Prior to the design of the plan representation, we felt it was critical to first record a glossary of EBO terminology, which was done with the help of Buster McCrabb, who contributed several white papers on explaining effects, indicators, and the overall EBO CONOPS [refs]. To maximize the reader's understanding of subsequent discussions, the most relevant definitions are included below:

Table 1. Glossary of EBO terminology.

Term	Definition
Effect	A state resulting from an action, which can result in an objective being achieved.
Task	A specification of an action to be performed. The action is the performance of a task.
Causal Linkage	A description of the cause-effect relationship between an effect or task and the effect it triggers, explaining how or why the effect should be achieved. Formerly known as Mechanism.
Objective	The clearly defined, decisive, and attainable aims toward which every military operation should be directed. Desired state or goal for an operation, independent of the consideration of an action.
Course of Action	A concept of joint air operations expressing what, when, and how they will affect the adversary or current situation.
Measure	A variable that characterizes a level of performance of a task or attainment of an effect, e.g. number of bridge spans destroyed.
Criterion	A predicate that can be applied to a measure, e.g. at least two X, or less than 20% of Y.
Standard	The application of a criterion to a measure, establishing an acceptable level of performance of a task, or attainment of an effect or objective, e.g. at least two bridge spans dropped.
Measure of Effectiveness (MOE)	A standard associated with accomplishment of an effect of a task.
Measure of Performance (MOP)	A standard associated with a force or a unit that specifies adequate performance of a task.
Indicator	A measure or standard that represents positive or negative evidence toward accomplishment of a task, effect, or a causal linkage.

Effects-based planning methodology evolved from and subsumes two previous planning methodologies. The first is objective-based planning. This method emphasizes bottom-up, target-centric planning. It is indicative of Cold War era attrition-based strategies in which the U.S. faced an equal superpower with similar capabilities and doctrine. The driving philosophy behind these strategies is to annihilate as much of the enemy's assets and forces as possible. This was later replaced by a more sophisticated methodology known as Strategy-to-Task (STT) planning. In STT, objectives at one echelon are decomposed into tasks and subtasks. These tasks then become the objectives of the subordinate echelon, and are themselves decomposed into lower-level tasks. The process continues, resulting in a plan representation nicknamed a "Z-Diagram" because of the zigzag shape of the plan when drawn in a tabular space with descending echelons as rows. STT was a significant step forward because it encouraged top-down styles of thinking based on objectives rather than bottom-up thinking based on destroying the enemy's targets. However, STT still lacked explicit representation of effects and the mechanisms leading to those effects. Put another way, it lacked traceability and explicit assumptions on how tasks would lead to their parent objectives.

Target-Based

- ID enemy entities, destroy them
- Focus: physical effects at target level
- Looks at 1st and 2nd order effects only
- No dynamic assessment
- No explicit timing considerations

Objectives-Based (Strategies-to-Task)

- Strategies at one level become objectives for next
- Focus: objectives at every level
- Considers linkages between objectives and strategies to achieve those objectives
- No dynamic assessment
- No explicit timing considerations

Effects-Based

- Address causality between actions and effects
- Focus: desired effects (physical and behavioral)
- Encompass both target and objective-based methods
- Models the enemy-as-a-system
- Considers many levels of effects
- Timing explicitly considered

Figure 1. Comparison of Effects-based, Objective-based, and Target-based approaches (source: EBO Overview briefing, EBO ATD kickoff)

Effects-based planning addresses this problem by decomposing objectives into specific statements of desired effects, typically consisting of an "effect verb" (including the canonical D11 verbs: destroy, disrupt, degrade, disable, deny, decapitate, delay, defend, deceive, deter, dislocate) and an object of the verb representing the enemy capability, action, or aspect against which the effect is to be achieved. EBO methods further transcend STT by explicitly considering the causal linkages, that is, the cause-and-effect relationships by which the planned task (action) leads to the desired effect. Obviously, these cannot be known with certainty, but by forcing planners to write them down explicitly, they will be better able to look for tell-tale indicators of these causal chains and will be more likely to correctly account for deviations from expected behavior during execution and repair the plan accordingly. Finally, EBO methods also

must include consideration of the timing of effects, including *delays* inherent in the causal linkages, and *persistences* explicitly representing the expected duration of temporary effects. Figure 1 illustrates how EBO planning methods subsume the previous STT and objectives-based methods.

Our initial design for the SDT plan representation was influenced by prior work on ground forces planning doctrine in which each effect (purpose) is explicitly paired with a corresponding task, which in turn could be decomposed into lower-level effect-task pairs. Figure 2 shows an example of this effect-task pair hierarchy. We later added a separate "Mechanism" construct as the causal link explaining the relationship between the task and effect in each pair.
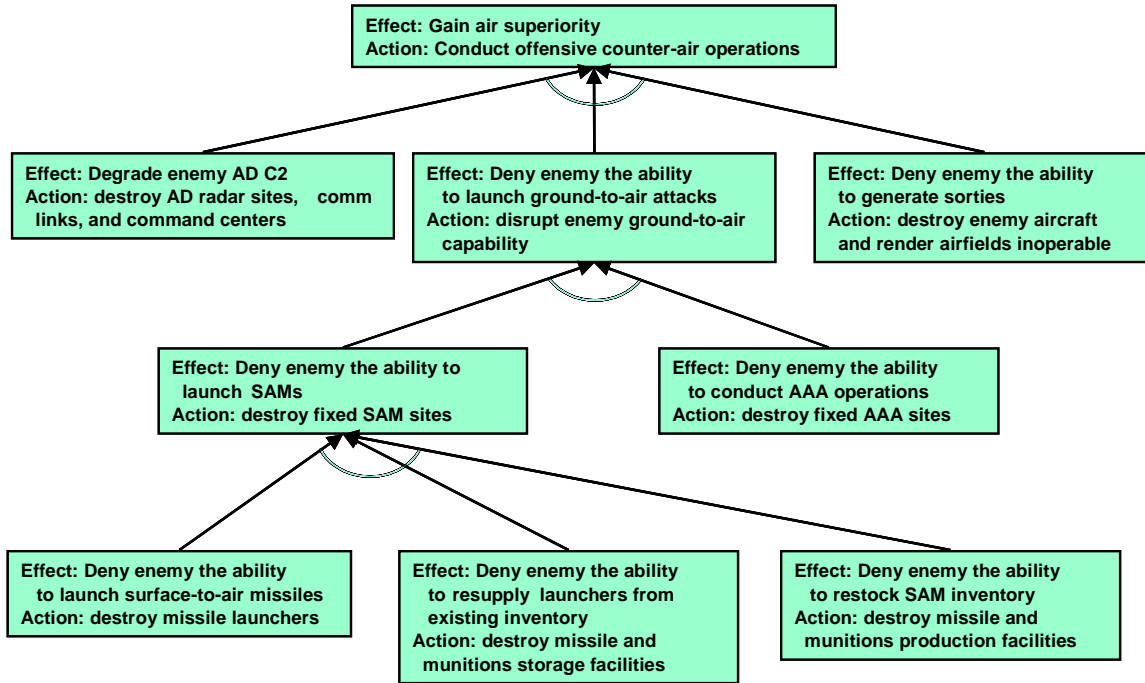
Figure 2. SDT's initial plan representation infused effects within an STT-like construct.

Another goal of this planning approach was to make explicit the mapping of blue (friendly) tasks and effects at each level to enemy Center of Gravity (COG) nodes within a pre-developed COG model. Here, we define a COG model as a directed acyclic graph in which the nodes represent enemy actions, capabilities, resources, conditions or general objects and the links represent cause-effect relationships of varying strength. Cross-COG models spanning an enemy's Leadership, System Essentials, Infrastructure, Population, and Fielded Forces, based on Warden's five-ring COG model (Warden, 1998) could be represented simply by labeling nodes according to this taxonomy and naming them according to some enemy aspect of interest within each COG. Figure 3 shows a fragment of a sample COG model from our original proposal.
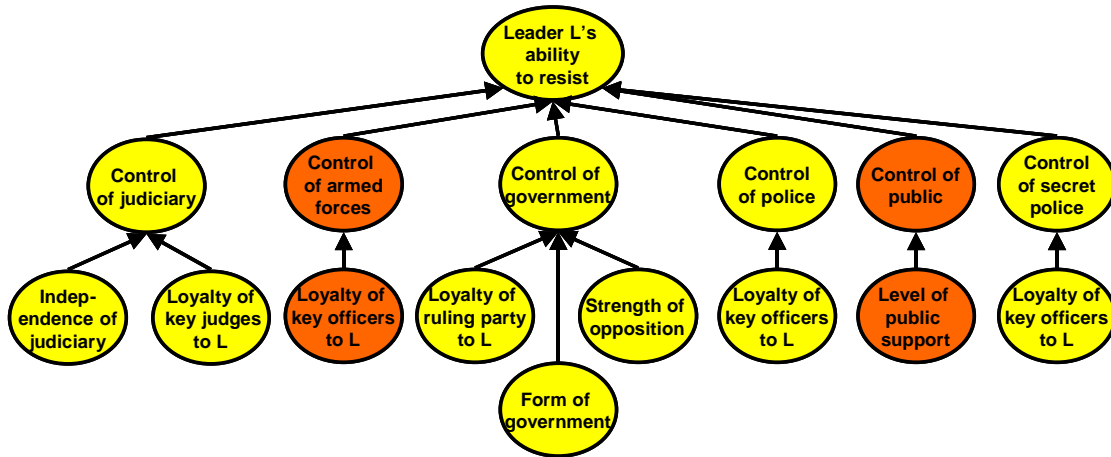
Figure 3. COG models capture causal relationships within and across enemy systems such as Leadership, Population, and Fielded Forces.

We expected that this approach would combine the strength of the STT style of planning, namely strong traceability between successive levels of planning, and the EBO planning methodology, because of its explicit representation of effects as independent from tasks. However, several months into the effort, as we began to formulate sample plans and work more closely with ISX and EBO subject matter experts, we encountered some significant shortcomings with this representation:

- It often led to redundancies between the task at one level and the effect at the next level. This blurred distinction between effects and tasks rendered the method equivalent to a simple hierarchical task network.

- Mapping this hierarchical effect-task construct to a coherent Bayesian causal model was problematic, since effects at one level could not "cause" tasks at the next level. They could instead be linked to their parent effect, but this would bypass the more direct influence of the parent effect's paired task.

- Our goal of mapping the plan to a pre-existing enemy COG model was more feasible, since blue (friendly) tasks could be said to inhibit a low-level red COG node, and blue effects could be tied to a particular state change (e.g. reduction in probability) of a downstream red COG node. However, the intermediate tasks in the construct again often proved redundant with the intermediate effects and would artificially augment the leaf-level tasks. In other words, only the leaf-level tasks were appropriate for inclusion in an explicit causal model, since higher-level tasks were simply an abstraction or aggregation of those original actions.

- The plan model was inherently different from the existing EBO ontology developed by ISX, which led to difficulties in reconciling our ideas into a uniform design for SDT.

The unified plan representation that arose was based more closely on the ISX ontology with some extensions influenced by the above scheme to accommodate mapping the plan to a causal model representation in CAT. Further extensions were made based on feedback during early JEFX '04 spirals. Figure 4 illustrates the hierarchical parent-child relationships allowed under this new plan representation using an UML-like graphical notation. All major plan element types inherit from the abstract base type *EboEntity*, which contains the fields *eboID*, a unique

8

identifier for the plan element, *eboName,* a user-defined short name, and *eboDescription,* a longer description of the element. The *Plan* element serves as the root node of the air component's specific plan to support the Joint Force Commander's (JFC) guidance. In addition to the inherited attributes from EboEntity, the Plan element includes a point of view (blue for friendly, red for enemy, and gray for neutral), and optionally a definition of C+0/L+0 (mobilization day/hour) and D+0/H+0 (deployment day/hour) in terms of absolute calendar time. D+0 may optionally be defined as relative to C+0. Planners may specify additional letter codes representing other key events in the operation, e.g. E+0, F+0, etc.
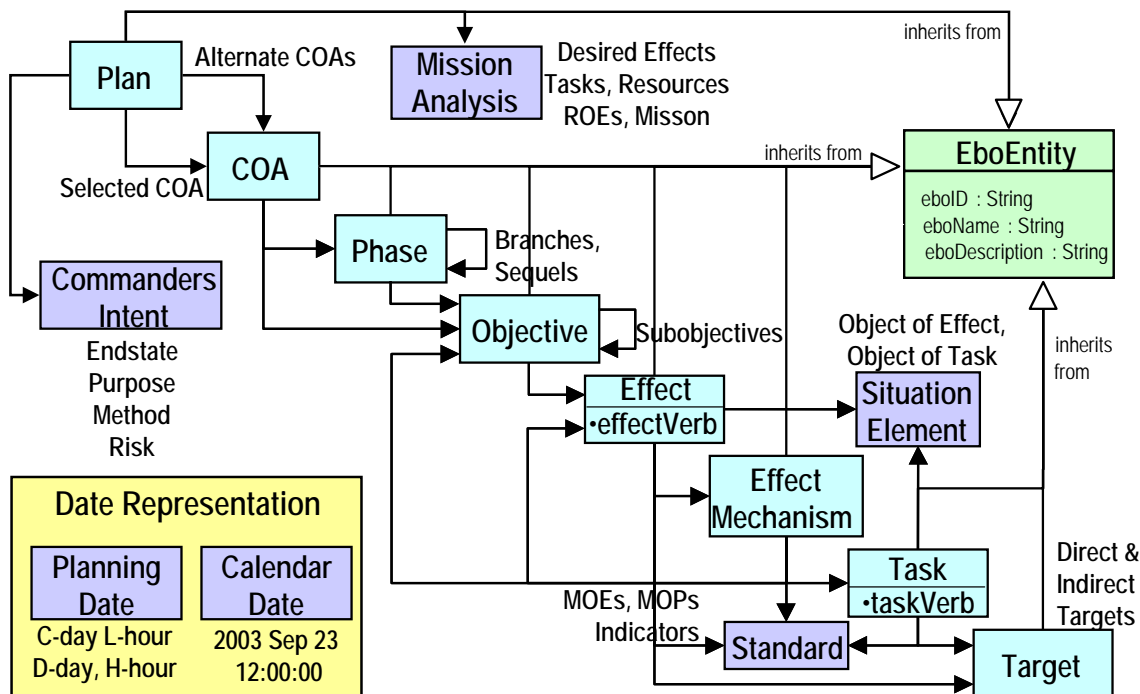
Figure 4. Final EBO plan representation for SDT.

Before fleshing out the plan, users enter *Commander's Intent*, itself composed of statements describing the JFACC's desired endstate, purpose, method and risk, and *Mission Analysis* (MA). MA consists of the following information elements, each of which may contain multiple statements unless otherwise specified:

- *Specified and implied tasks* for the operation, with some relatively small subset of these designated as *essential tasks.*

- Available *resources*

- *Rules of engagement*, which consist of:

  - *Constraints* that must be performed and

  - *Restraints* that must be avoided.

- *Facts*

- *Assumptions*

- *Commander's Critical Information Requirements*, which consist of lists of these items:

  - *Type* (Enemy, Friendly, Environment, or Other)

  - *Priority* and *Weight of Effort*, specified either as a percentage or label (low, med, high).

  - *References* – recommended references for obtaining the desired information.

  - *Description* – a description of the requested information.

- *Mission Statement* – A succinct paragraph describing the goals of the operation.

The root Plan is decomposed into one or more *courses of action* (COAs) representing alternative approaches to achieving those goals. Eventually, after subsequent activities of COA analysis and COA comparison, the most desirable COA is designated as selected for execution. Optionally, a COA that is complex and occurs over a significant period of time may be broken down into multiple *Phases*, which are one of three types:

- *Initial* phase – begins the operation

- *Sequel* phase – Phase following another phase when that phase's objectives are obtained

- *Branch* phase – Phase triggered by occurrence of an event specified in a *branch condition,* only valid during an associated previous phase.

The phases of a plan form a tree structure in which the actual path taken during execution represents a single path starting at the initial phase, proceeding with a string of sequels and branches until a phase with no sequels or branches is reached (notwithstanding changes to the plan during execution).

Within a Phase, or optionally directly within a COA, the plan specifies one or more *Objectives*, each representing a distinct desired state of the operation. As a rule of thumb, objectives tend to be stated positively from the blue side's point of view, e.g. "Gain air superiority." In order to be backward compatible with previous planning methodologies, objectives may be directly decomposed into subobjectives, and may have one of three successively lower levels: Strategic, Operational, and Tactical. Typically, tactical objectives may be considered as synonymous with desired effects. In addition, objectives have the following attributes: *name, description, priority, weight of effort* (these last two are used as resource considerations for the subsequent activity of Guidance, Apportionment, and Targeting), and Measures of Effectiveness (often called Strategic Indicators at this level).

Each objective in turn may be decomposed into one or more desired or undesired *Effects*. In contrast to an objective, which is an open-ended statement, an effect is comprised of two main parts: an *effect verb,* such as the "D11" (destroy, disrupt, degrade, deny, etc.), and an *object of effect,* which is the enemy aspect being affected. Also, desired effects, unless of a defensive nature, tend to be stated as acting against the enemy. Optionally, an effects statement may contain an *indirect object of effect,* which in the case of SDT, is used when the direct object of effect is a commodity such as electric power to represent the intended recipient of the commodity (see Figure 5). Desired effects also contain some of the same attributes as objectives, including *level, priority,* and *weight of effort,* and *Measures of Effectiveness.* Attributes special to effects include desired start date/time (specified as a C/D date or calendar date), desired overall duration, location, and indicators (see Table 1 for definitions of MOEs and

Indicators). In order to take full advantage of the temporal analysis capabilities of AFRL's CAT, we added the properties *delay* and *persistence,* time durations representing the inherent wait period before an effect manifests (even after its causal linkage has been fulfilled), and the period that the effect is sustained (i.e. time needed for the enemy to work around or neutralize the effect). Prior to JEFX '04, the plan representation was extended to include *undesired effects,* which contain the same attributes as desired effects, but are intended to represent potential negative effects that occur as a result of the subordinate tasks and effects.
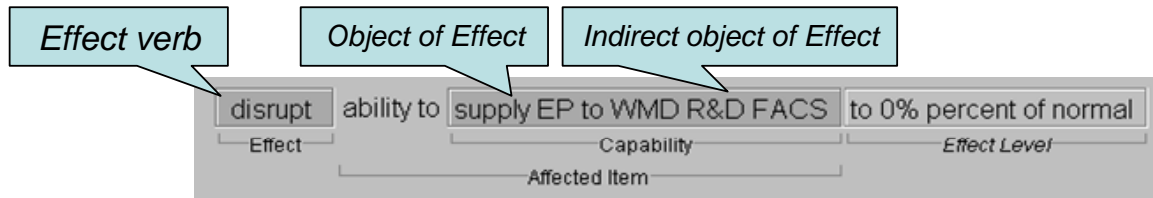
Figure 5. Sample desired effect statement.

The next two levels of decomposition are *causal linkages* (previously known as *mechanisms*) and *Tasks*. Effects are only linked to tasks via causal linkages, which provide an explanation of the causal mechanism (the how and why) set in motion by the actions taken in the task eventually culminating in the effect. Causal linkages become especially important for effects-based assessment of plans during execution. By making explicit the assumptions about the intermediate process connecting a task (cause) and its effect, planners can more readily adapt their plans when there is positive evidence of task achievement but negative evidence against effect achievement. In addition to textual attributes (short name, description, explanation), a causal linkage has a *causal strength,* an estimation of the likelihood of the effect's achievement, conditioned on successful task performance. Like effects, causal linkages may also have a delay and persistence, representing how long the intermediate causal process needs to complete and how long it is sustained. Causal linkages may be used not only to explain direct effects arising from actions, but secondary or *cascading* effects arising from intermediate effects.

Tasks are typically the lowest level elements of the plan before it is passed on to targeteers. They represent specific blue actions to perform. Like effects, they contain a *task verb,* from a set of action-oriented verbs overlapping partly with the effect verb list, and an *object of task,* the enemy aspect on the receiving end of the action. Tasks share the attributes *level, priority,* and *weight of effort* with effects and objectives. Like effects, they have a desired start date/time and duration, location, Measures of Performance, and indicators. One attribute special to tasks, *success probability,* was added in order to map to the leak (prior) probability of the CAT representation of the task. Tasks may optionally be decomposed into Target nodes, representing either an individual uniquely identified target or a target set representing all targets of a particular type in the area of interest. Targets were added to SDT's plan representation in order to support effects-based target systems analysis, described later in Section 3.4.

3.2   SDT PLAN EDITOR

This subsection chronicles the development of the SDT Plan Editor, the primary plan authoring tool for specifying effects-based plans during COA development. For the first couple years, the team used an evolutionary prototyping process in order to obtain early feedback from SMEs and establish feasibility of integration with analysis tools, weaving refinements based on

preliminary findings directly into successive iterations. In the outyears, we adopted a spiral development process aligned with customer milestones and JEFX '04 spirals, during which our efforts were focused on maturing established capabilities and developing new capabilities based on well-specified requirements from Government SMEs and JEFX participants.

The first major Plan Editor prototype embodied the hierarchy of Effect-Task pairs described in our original proposal, shown graphically in Figure 2. As shown in Figure 6, the application menubar, shown just below the title bar, and the toolbar shown just below that, served as redundant methods for opening or saving a plan to XML, or performing edit operations on a node, such as new child, cut, copy, delete, or paste. The left-hand pane utilized a tree view to depict the hierarchy of plan nodes. The right-hand side, covering the majority of the screen area, contained the attribute details of the currently selected node in the tree. Read-only attributes include "Supports" and "Supported by," which show the names of the parent and child plan elements. Editable attributes include a short Name summarizing the goal of the plan node, and a Mechanism describing how and why the specified task leads to its paired effect. Within the Desired Effect and Task regions are a series of expandable statements describing the Effect or Action, where it should occur, when it should occur, the red COG node representing the object of effect/task, and an optional Measure of Effectiveness.
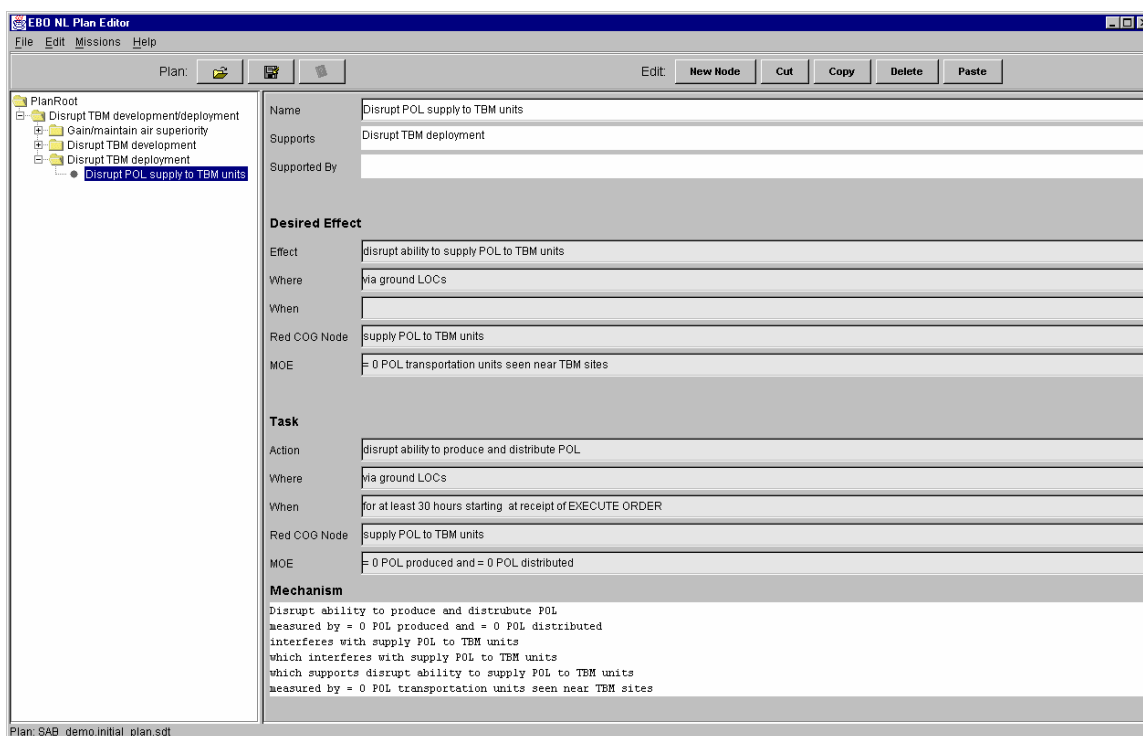
Figure 6. Early prototype of SDT Plan Editor using effect-task pair hierarchy.

Clicking on the field showing the statement text expands the field into a restricted natural language template. The template automatically guides the user in correctly entering the statement's constituent phrases. The purpose of each phrase is shown in the bracketed label below the phrase. Each phrase shown within a bracket-labeled button is itself a mini-template, which expands when the user clicks on the phrase. For example, in Figure 7 the Affected Item phrase has just been expanded to a mini-template starting with the fixed text "ability to" and ending with a user-specified Capability. Dark gray sections are required to be filled out, while

others, such as the Effect Level, are optional. Eventually, all templates bottom out at either fixed text, a drop-down menu of set choices, or a free text field.

.
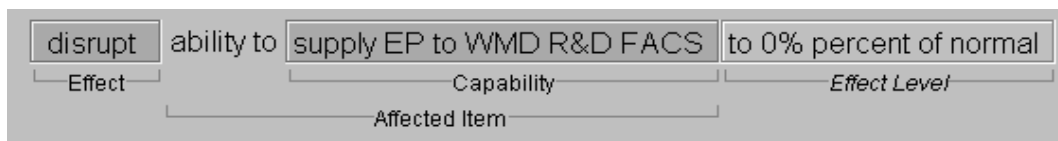
Figure 7. SDT's restricted natural language templates guide users in creating Effect statements that format effect information for consumption by analysis tools.

The rules governing the construction and dynamic expansion of these restricted natural language UI templates were originally specified in a grammar file compatible with XSB Prolog's Definite Clause Grammar notation. The original version of the restricted NL UI software inherited from our Active Templates project was implemented entirely in Java, and failed to adequately separate the grammar rule representation from the user interface management code. Later, a new natural language engine was developed in XSB Prolog that took full advantage of Prolog's built-in natural language parsing engine, eliminating the unwieldy Java-based rule engine.

Aside from the user-oriented advantage of having built-in guidance in entering plan statements, an expected developer-oriented advantage was ease of extracting relevant information from portions of the statement's parse tree. Unfortunately, the original API for the NL UI required keeping the full path of bracketed labels to obtain a particular phrase of interest. This complicated the process of mapping key subphrases to plan element attributes in the Plan Editor's central Java data model. This mapping became a critical need as we started to integrate ISX's externally developed plan editor views. After the initial round of integration, we identified a more serious shortcoming in the API, namely, lack of ability to generate parse trees from statements when the grammar contains free-text fields.   This was needed to take assembled statements from the ISX data model (such as effect.verb + effect.objectOfEffect) and turn them into parse tree structures that could be understood by SDT's NL UI. Our solution to this problem was to add another layer of information to the templates, called semantics. Semantics are named attributes embedded within the grammar notation. The Prolog-based NL engine was revised to automatically "fill in" values of these attributes based on the associated parse tree node, making the attribute value available to Java by a simple lookup. This significantly eased the process of extracting and mapping information from the templates to populate objects. The reverse process of recreating NL templates from the combined ISX/SDT objects was also simplified, though it proved somewhat brittle as the SDT's plan representation evolved rapidly.

As SDT was exposed to a wider range of users prior to its acceptance in JEFX '04, we gradually came to realize that the relatively low complexity of the statements required for the plan node attributes did not justify the sophistication of a UI design based on restricted NL templates. The significant number of extra clicks required to fully expand the templates was seen by many users as cumbersome compared to simply typing in entire sentences freehand (a longstanding goal for the NL UI that lost out to tasks of higher priority). However, completely freehand statements for plan nodes would defeat our goal of being able to extract information in specific phrases in order to populate the object-oriented plan representation. Accordingly, we migrated to a simpler UI design that satisfied both goals. Each plan node would have a Name

field where users could enter a freehand short description of the element. But below that would appear a series of labels and corresponding fields laid out much like a web-based data entry form, where each field was directly mapped to an object attribute. Figure 8 shows an example of this layout, with the root Plan node editor pane showing the Name field and below it the Description field.

Many cosmetic improvements and built-in capabilities were developed on the three-year road to the final version of the Plan Editor shown in Figure 8. The generic tree view in the original prototype was tailored to use intuitive icons representing different plan element types, and the tree view was enhanced with a subordinate toolbar allowing users to do a full expand or collapse of a node, and navigate forward or backward in the node selection history. The right-click menu of the Tree View was expanded to allow users to add nodes as a *parent* of the selected node, allowing users to plan bottom up instead of strictly top-down when the editor was limited to adding child nodes. A new Undesired Effects tab was added to the Tree View, for specifying Do Not Strike targets and Do Not Affect target restrictions underneath each COA (see Section 3.4.3).
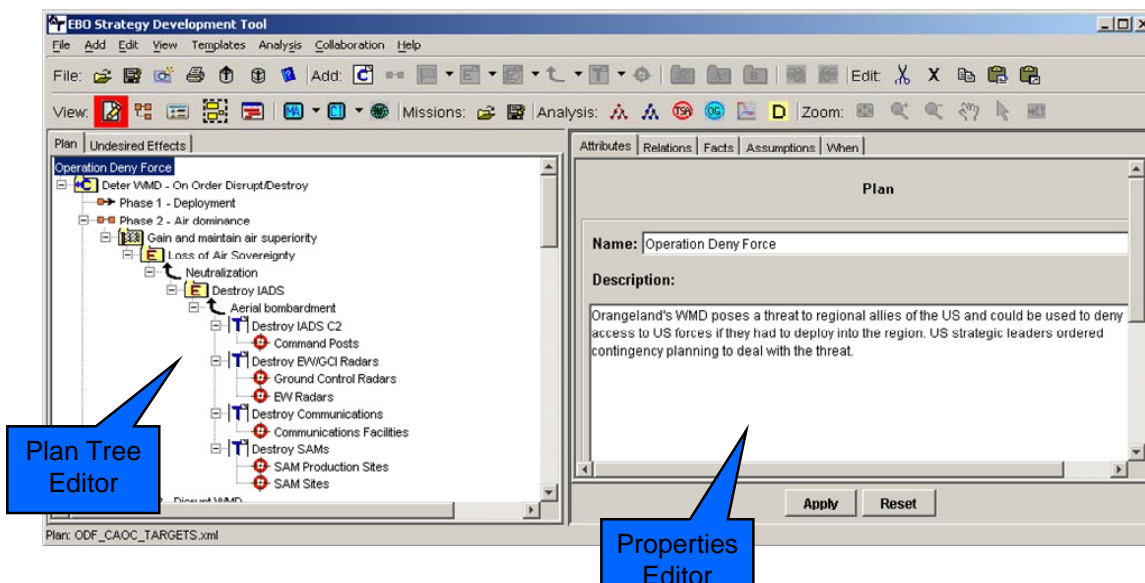
Figure 8. Final SDT Plan Editor and demonstration plan spanning all plan element levels.

The *File* menu was expanded to include actions to snapshot the current editor window as an image, print the window, and import or export the plan to the TBMCS Strategy Management Service. The Add toolbar, like the Tree View, was extended to allow adding nodes as parents or children. The View toolbar now includes many alternate views for the right-hand side of the Plan Editor: Properties View, Timeline, Structural Editor, Target Set Tool (map view), as well as Commander's Intent and Mission Analysis dialogs for either the Joint Force Commander (JFC) or Joint Force Air Component Commander (JFACC) level of planning. A *Missions* toolbar was added to allow users to insert a mission template, a pre-defined plan fragment corresponding to a canonical Air Force mission, under the selected plan node, and to save the portion of the plan starting from the selected node as a new mission template. The Analysis toolbar provides actions for launching the COG Articulator (see Section 3.5), exporting the blue COA to COG Articulator and OAT, launching Endstate's Target System Analysis or Option Generation algorithms, viewing the Endstate outage profile graphs, or launching the DCOAD tool. Finally,

the Zoom toolbar was added to allow pan, zoom, collapse, and expand operations within the Timeline and Structural editors, which were improved incarnations of the original ISX Phase, Objective, and Timeline views.

These enhancements are described in more detail in the following subsections.

3.2.1 Mission Templates

Mission Templates are tactical-to-operational level reusable plan fragments encapsulating canonical Air Force missions. Implementation of mission templates consisted of two main tasks: 1) implementing the capability within the Plan Editor to save and load mission templates, and 2) authoring specific mission templates based on doctrinal sources.

Implementation

During the first spiral of Plan Editor development, we implemented the baseline functionality for Mission Templates under the *Templates* menu/toolbar. When the user chooses the *Save as Mission Template* menu item, the plan fragment starting from the currently selected node in the Tree View is saved to an XML file under a special subdirectory called "templates." When the user selects the *Insert Mission Template* menu item, a file chooser dialog appears listing all available mission templates in the "templates" directory. The user may select a template, click "Open," and the system inserts the plan fragment under the currently selected node in the Tree View, as illustrated in Figure 9. Originally, this root insertion location could be at any level in the plan, since all plan nodes used the same action-effect pair construct. As the pattern editor migrated to a top-down ontology of different plan node types, we added a validation step to the *Insert Mission Template* action that ensures the root node of the plan fragment is a valid child node for the currently selected plan node in the Tree View.
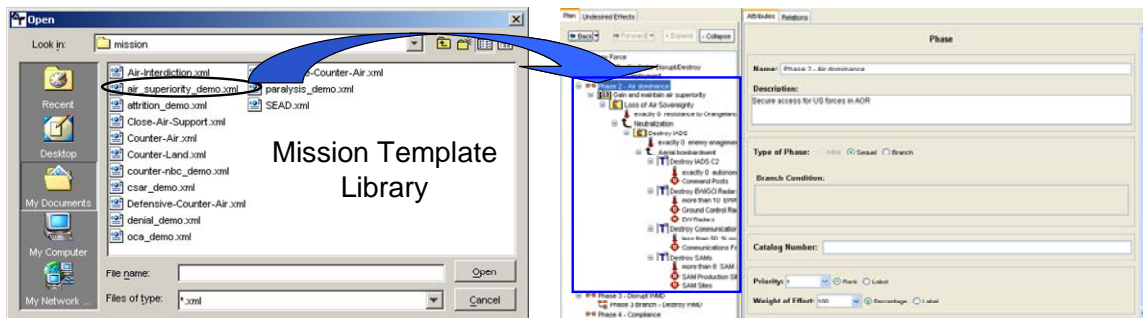
Figure 9. Mission templates are reusable plan fragments based on canonical AF missions.

Template Authoring

Initially, a small set of about five mission templates were developed by the BAE Systems team, based on canonical doctrinal Air Force objective/task decompositions in AFDD 2-1 and the UJITL (2002). Interior (non-leaf) tasks/objectives were usually translated into effect nodes and plan node statements were paraphrased to fit the [effect/task-verb, object-of-effect/task] NL UI templates. Later, after the NL UI templates were replaced with conventional user interface forms (i.e. labels, textfields, radio buttons and choiceboxes), AFRL commissioned BlueForce LLC to develop a broader suite of detailed mission templates, including MOEs and MOPs. As with the original suite, context-specific information such as "Where" and "When" were omitted

to allow for tailoring of templates to particular planning situations. The full list of mission templates is as follows:

- Air-Interdiction.xml – Air operations conducted to destroy, neutralize, or delay the enemy's military potential before it can be brought to bear effectively against friendly forces at such distance from friendly forces that detailed integration of each air mission with the fire and movement of friendly forces is not required.

- Airlift.xml – Transport personnel and equipment through the air.

- Close-Air-Support.xml – Prevent opponent from dominating the surface environment and ensure the success or survival of surface forces.

- CSAR.xml – (Combat Search and Rescue) - Contingency mission for search and rescue of downed friendly aircraft and airmen during combat operations.

- Counter-Air.xml – Enable friendly operations against the enemy and protect friendly forces and vital assets through control of the air.

- Counter-Land.xml – Rapidly destroy or render ineffective significant portions of key enemy surface forces and their supporting infrastructure, thereby enhancing maneuver warfare and avoiding prolonged and costly wars of attrition.

- Defensive-Counter-Air.xml – DCA operations defeat enemy air threats after launch. DCA operations are conducted at three levels: area defense, point defense, and self-defense.

- Offensive-Counter-Air.xml – Includes subordinate missions of surface attack, fighter sweep, escort, and suppression of enemy air defenses.

- SEAD.xml – Suppresion of Enemy Air Defenses. Any activity that neutralizes, destroys, or temporarily degrades enemy surface based air defenses by destructive and/or disruptive means.

- Strategic Attack.xml – Operations intended to directly achieve strategic effects by striking directly at the enemy's centers of gravity (COG).

3.2.2   Strategy Templates

During the second year, we worked with ISX Corporation to integrate their *strategy templates* capability, prototyped within their *Effector* planning tool. While mission templates focused on tactical-to-operational level plan fragments, strategy templates were intended to be strategic-level skeletal plan outlines embodying a particular overarching approach to an operation. Moreover, while mission templates are authored using the same plan editors as regular plans, ISX's implementation of strategy templates used a methodology that requires users to employ special *template nodes* for effects and tasks in which the verb is specified, but the object-of-effect/task is a *situation element type* rather than a specifically named target or object. These situation element types are defined in ISX's situation element ontology, which is loosely based on the Modernized Integrated Database (MIDB) category codes and other related taxonomies. Strategy templates are saved to a special folder (templates/strategy/*.xml) and later may be *instantiated* via a template instantiation wizard. The wizard walks through all effect-templates and task-templates in the skeletal outline and prompts the user to browse a knowledge

base of situation element *instances,* restricting the list to instances of the given situation element type. After all template nodes are converted to concrete effects and tasks against specific objects, the template has been turned into a basic plan, with less effort than would be required to build the plan from scratch.

This capability was integrated within SDT 1.2 as an internal function of ISX's Objective Editor. The user would enter a special mode for template authoring via the *Templates* menu. Then all subsequent effect and task nodes added to the plan would prompt the user to choose a situation element type as the object of effect. This was done via a browser that restricted the element types based on the choice of verb; these restrictions were part of the situation element ontology encoded in DARPA Agent Markup Language (DAML). As BAE Systems prepared for JEFX '04 and replaced the Objective Editor with the Structural Editor, this capability was not retained, partly because of the additional authoring complexity, and partly because the process of populating the situation element ontology was redundant with pre-existing Intelligence Preparation of the Battlefield (IPB) tools and processes already included in JEFX '04. However, the basic strategy template concept was retained by converting all original ISX strategy templates to regular skeletal plans in SDT's format. Both the three strategy templates authored by Buster McCrabb (attrition.xml, denial.xml, paralysis.xml) and the IW strategy templates jointly developed by ISX and Adroit Systems (now SRA) are retained in SDT under the templates/strategy/ and templates/mission/iwtemplates/ folders.

3.2.3   Effect Dictionary

The *Effect Dictionary* was a small but significant component developed during the first year to give users a quick reference to look up definitions of canonical effect verbs and view sample effect statements. The definitions were based on an Excel spreadsheet filled in based on multiple exchanges with our SME, Buster McCrabb. To enforce the notion that effect verbs can be applied to a range of centers of gravity (COG) and types of affected objects, multiple sample statements were included for the same verb, based on the user's selected affected item and COG. Figure 10 shows an example of the definition and sample statement for disrupting a capability within the infrastructure COG.

During JEFX '04 Spiral 1 we received user requests to post the doctrinal source for the effect definitions. Their motivation was to understand whether the definitions were Air Force specific or valid for joint operations. Future SDT releases included a label at the top citing AFDD 2-1 and EBO subject matter experts as the sources. In preparing for JEFX '04, we made the effect dictionary more extensible by migrating the effect dictionary rules from Prolog code to a lightweight XML representation, in resources/effect_definitions.xml.
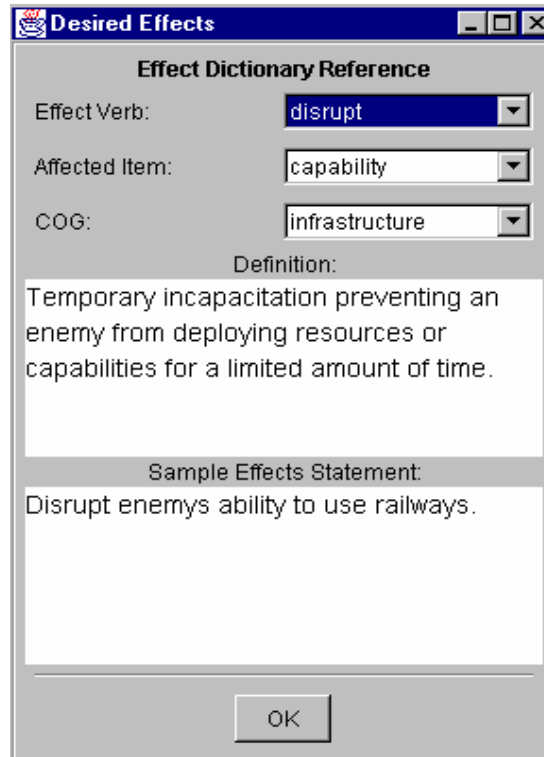
Figure 10. The Effect Dictionary provides definitions and sample effects statements for different COGs and affected objects.

3.2.4 Timeline Editor

The Timeline Editor, shown in Figure 11, is an alternative Gantt-style view of the plan that depicts the start and end times of all plan elements relative to calendar time or operation time (C+0 and D+0). The left side is similar to the Tree View, showing the hierarchical structure of the plan using indented icons and labels and collapsible nodes. To enhance the readability of the schedule, causal linkages are omitted. Immediately to the right of the tree pane are several table columns conveying the start date, end date, and duration of the plan element. Start date and end date are given relative to C+0 here. Finally, the righthand side comprising the majority of the screen real estate shows the schedule with time increasing to the right. One Gantt bar appears per each plan element, aligned with the element's table row. The schedule headers may be tailored to show multiple timescales, including C+0, D+0, and calendar dates. The schedule view may be tailored via the "Layout" menu/toolbar, to zoom in or out or layout the schedule to fit in the available window space.

The initial versions of the Timeline Editor were developed by ISX as part of their *Effector* tool suite. It was delivered to the BAE team as a component in the form of a Java *jar* file and corresponding *Javadoc* API documentation. In SDT 1.2 we added a new View button for toggling to the Timeline Editor and wrote code to translate from our internal plan data model to the ISX data model required by the Timeline Editor. Unlike *Effector*, this translation was unidirectional, such that changes made in SDT would be reflected in the timeline but not vice versa. We investigated declarative bidirectional translation approaches but maintaining them was determined to be too costly given the frequency of changes to both data models. For SDT

1.4 we adopted a data model based more closely on the ISX ontology data classes, but using an SDT-centric Java interface package (*com.alphatech.ebo.rep)* and SDT-specific implementations that inherited from the ISX core classes. This approach yielded three primary benefits:

1. It allowed us to use a single data model instance, avoiding costly translation

2. It supported bidirectional interoperability between ISX and BAE editors

3. It enabled us to rapidly extend the ontology classes to accommodate new features, helping us keep up with the rapid requests for new features arising throughout JEFX '04.
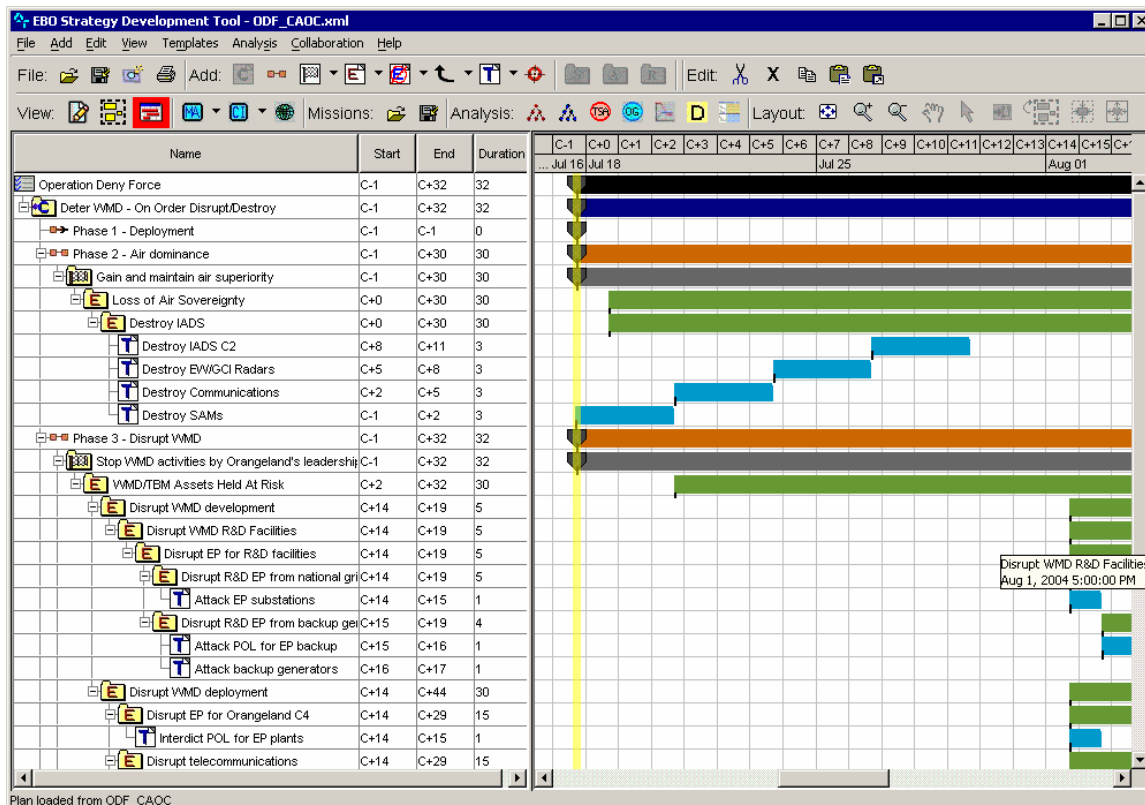
Figure 11. The Timeline Editor is a Gantt-style view of plan element times and durations.

After JEFX '04 Spiral 2 we received end-user enhancement requests to the timeline that would not be possible without source code for the Timeline Editor, and direct support from ISX was no longer feasible. Thus, with AFRL's permission, we developed a new Timeline Editor for Spiral 3 designed to support the additional features. We utilized the same ILOG JViews 5.5 COTS tool used by ISX, which we had already purchased for development of COG Articulator (see Section 3.5). This allowed us to rapidly reconstruct baseline features and develop key extensions:

- Independent Gantt bars for tasks and effects since effects are typically temporally downstream (effects were rollups of tasks in the original implementation)

- Different color Gantt bars for each plan element

19

- Additional timescales for calendar dates and D-dates

Additional enhancements were developed for JEFX '04 Main Ex, including schedule assessment features such as a vertical current-time marker, and completion bars within Gantt bars denoting progress toward completion of the plan element. These changes are described in more detail in Section 3.7.3.

3.2.5   Structural Editor

The Structural Editor (shown in Figure 12) followed a similar pattern of development to the Timeline Editor. It was also prototyped by ISX within *Effector* as the primary plan view, shown alongside a plan tree view. It was integrated into SDT 1.2 at the same time as the Timeline Editor as an alternative to the main SDT Plan Editor view (now called the Properties Editor). It actually was comprised of two distinct views, an Objective Editor and a Phase Editor. The Objective Editor graphically displayed the plan fragment structure within the selected objective in the Tree View, using different colors and shapes for effects, mechanisms (now called causal linkages), tasks, and sub-objectives. The Phase Editor showed a Pert-style dependency graph of the branch/sequel phasing structure of the plan. Like the Timeline Editor, integration with these views was unidirectional in SDT 1.2 in that changes made in SDT native views were reflected in the Objective Editor but not vice versa. In SDT 1.4 our migration to a unified data model based on the ISX ontology achieved bidirectional interoperability, but we were still unable to make enhancements to the view itself without access to source code.

Thus, for JEFX '04 we developed a new Structural Editor that combined the capabilities of the Phase and Objective Editors. Extended features included:

- Distinguishing branch from sequel arrows by color

- Collapsing/expanding phase nodes to show internal plan structure

- Support for both top-down (add child) and bottom-up (add parent) planning styles (the original version was strictly top-down according to the EBO core ontology)

- "Strategy to Task" mode for freeform planning with no parent-child type restrictions

- Double-click to bring up a floating Properties Editor dialog for a plan node

The experience gained from using JViews 5.5 to develop COG Articulator in 2003 allowed us to successfully prototype this view in the short period between JEFX '04 Spiral 2 and Spiral 3.
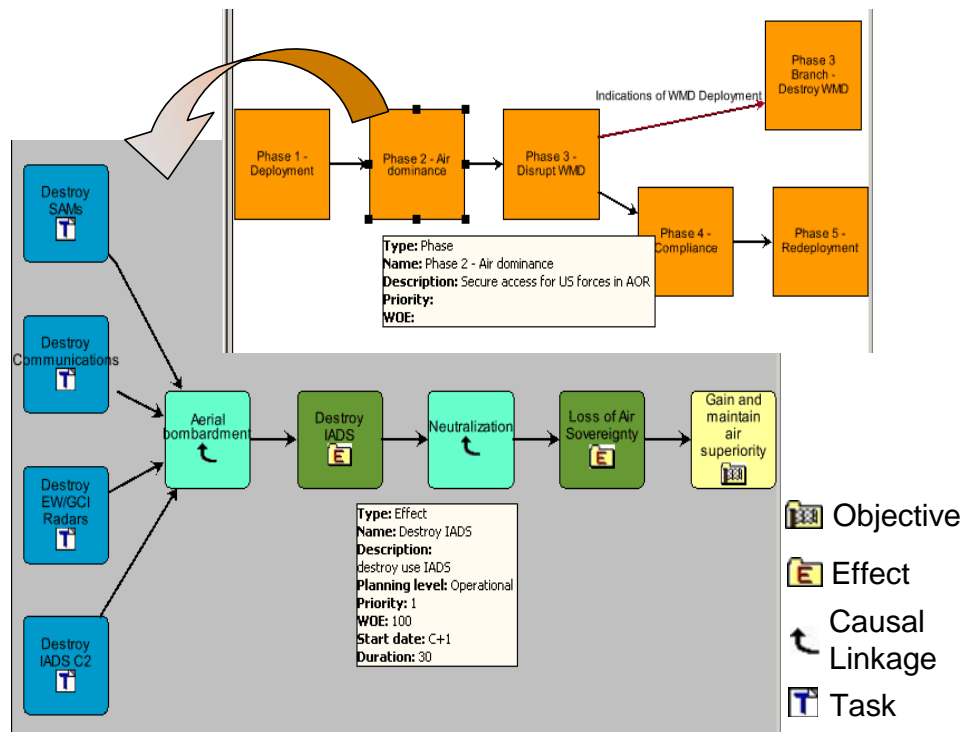
Figure 12. The Structural Editor enables bottom-up and top-down planning using hierarchical drilldown from COA phase structure to internal causal structure of objectives.

3.2.6 Mission Analysis and Commander's Intent

During the first year, we developed a simple Properties Editor for Commander's Intent (CI) statements using conventional text areas for Endstate, Purpose, Method, Risk, and Mission Statement. These fields appeared in the Properties Editor when the user selected the root plan node. No NL UI template was developed because these are such open-ended information items and they are often cut-and-pasted from the Joint Force Commander's Warning Order and tailored for the JFACC's Intent. Additional tabs for Mission Analysis (MA) were not included because the SDT's main focus was on COA Development.

However, as we approached JEFX '04 it became clear that users would need a more comprehensive editor for MA based more closely on the Joint Air Estimate Process described in JP 3-30 (2003). To this end, in SDT 1.3 (mid-2003), we designed and prototyped a new set of editors for MA and CI, each with a series of tabbed panes for entering one or more statements about a particular information category. To maximize code reuse, we developed a GUI component library of single- and multi-statement dialogs and adopted a declarative configuration scheme for defining tabbed pane names and mapping of fields to data objects. We also built reusable model-view-controller pattern classes that could be rapidly tailored for each editor. Finally, to ease import of text from other MS Windows applications, we added Cut/Copy/Paste context menus to the text fields. Direct drag-and-drop of selected text from other applications such as MS Word or Powerpoint was also supported.

By this time the root ComponentPlan node had evolved to include global information such as name, description, point of view, and timing information, so we moved the access point for MA and CI to new buttons in the View toolbar. CI statements, consisting of Endstate, Purpose,

21

Method, and Risk, could be entered into two separate dialogs, one for JFC-level and another for JFACC-level. The MA information consisted of the following tabs:

- Specified Tasks – Tasks spelled out in the Warning Order

- Implied Tasks – Tasks implied by the Specified Tasks

- Essential Tasks – The subset of the Specified and Implied tasks that are on the critical path to the Mission objectives (see the checkboxes in Figure 13)

- Resources – A list of resources that are to be made available to the commander

- ROEs – Rules of engagement, consisting of *constraints,* required actions, and *restraints,* prohibited behaviors

- CCIRs – Commander's Critical Information Requirements – A list of data or information that will be necessary to produce an accurate plan

- Facts – A list of known facts about the operation

- Assumptions – A list of assumptions that will be in effect for the plan

- Mission – A concise statement describing the primary goals of the operation

The final editors for MA and CI are shown in Figure 13. In preparation for JEFX '04 Main Ex, we extended these editors to support real-time collaboration via tab-by-tab locking. This work is described in more detail in Section 3.7.2.
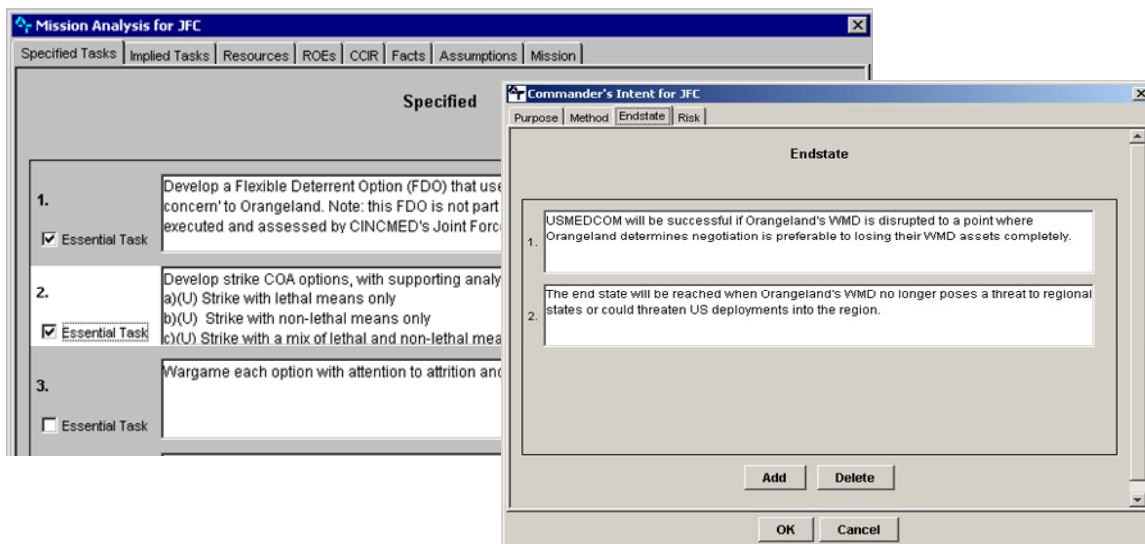
Figure 13. SDT uses tabbed panes for multiple categories of statements for Mission Analysis and Commander's Intent.

3.3   COA ANALYSIS USING CAT

This section describes BAE Systems' initial integration between SDT and CAT, allowing CAT to be used to provide a feed forward prediction of the SDT COA's probability of success

over time. All of the structural, temporal, and probabilistic inputs to CAT are derived directly from the plan elements and attributes entered in the SDT for the blue COA. The main purpose of this type of assessment is to ensure that the local probability and temporal estimates in SDT, such as causal strengths, and delay and persistence durations specified on parent and child plan elements, lead to favorable peak probability levels and timing for higher-level effects and objectives. Furthermore, in early EBO ATD experiments, translating the COA as a dynamic Bayesian net in CAT enabled further elaboration of the COA by downstream targeting and scheduling tools.

Before we proceed, it is important to note that the terms "parent" and "child" have different meanings depending on whether they refer to SDT plan elements or nodes in a CAT dynamic Bayesian network. In the SDT, the parent is the higher-level plan node and the child elaborates the parent, as an effect does for an objective. In CAT, the parent is the origin of a causal linkage that causes or inhibits a child, as with a task and its downstream effect.

3.3.1   Structural Mapping

By structural mapping, we refer to the topological translation of the COA hierarchy in the SDT into a graph of actions and causal linkages in CAT. In feed forward assessment, this mapping is straightforward, as the graph structure in CAT closely mirrors the plan structure in the SDT. The translation begins at the root of the plan, which is the COA object. This becomes the top-level node in the CAT Bayesian net model. That is, there are only incoming and no outgoing causal linkages on this node, and all other nodes lead to this node via some chain of causal linkages. The SDT then traverses the plan hierarchy in depth-first fashion, translating each plan element into a CAT node, and adding a causal linkage from the plan element to its parent from the SDT. Thus, the objective and effect nodes in the COA become interior nodes in the CAT Bayes net. The translation ends with the task nodes in the COA, which become the root, or lowest-level actions in the CAT model. That is, they have only outgoing and no incoming causal linkages. Figure 14 illustrates how a typical COA is mapped to a CAT model.
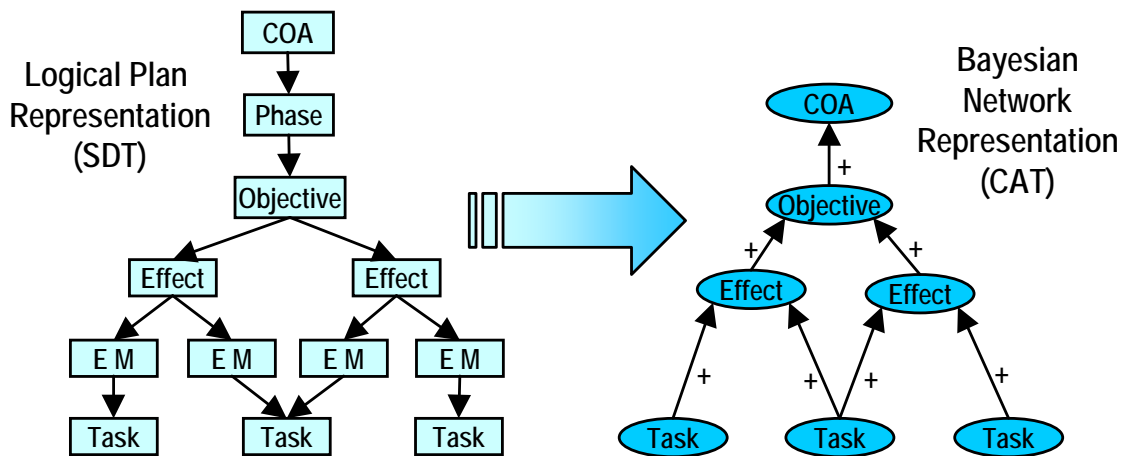
Figure 14. SDT translates the logical plan representation to a Bayesian network for analysis in CAT.

There are several notable exceptions to the one-to-one correspondence between SDT plan elements and CAT actions in the translation algorithm:

1. In the SDT, causal linkages are represented as first-class plan elements with their own icons in the tree view. These are translated directly into mechanisms in CAT, representing causal links between tasks and effects, or effects and subeffects. These are the only plan elements that translate directly into CAT causal linkages; for other types, a new CAT causal linkage is created to link a child plan element to its parent.

2. In the SDT, MOEs and indicators on tasks and effects are not shown in the Structural Editor (though they optionally may be shown in the Tree View) but are entered in the Properties editor pane for these plan elements. However, each MOE (and indicator that specifies a criterion) is translated into CAT as an MOE (evidence) node with a causal linkage pointing from the task or effect to the MOE.

3. Since branch phases represent different conditional paths through a COA, the SDT currently prompts the user to choose a subset of Phases to translate. By default, the SDT translates the "main" branch of phases by following the chain of sequels, starting with the COA's initial phase. Note that Phase elements are not translated into CAT actions, but their probability may be equated with the probability of their constituent objectives.

To clarify which actions were derived from which plan elements in the COA, the name attribute displayed in the SDT tree view is shown as the CAT node name, the upper node label in the CAT plan view. Also, the plan element's type (COA, objective, effect, or task) is used as the lower action label appearing under the name.

3.3.2   Temporal Mapping

CAT provides three main types of temporal attributes that may be entered as inputs to actions or mechanisms: scheduled probabilities, delays, and persistences. Scheduled probabilities are typically placed on the root nodes as triggers for processes modeled by downstream nodes. Delays and persistences represent waits or holds on probability levels from parent influences. Delays are typically used in interior CAT nodes or mechanisms, but persistences may apply at all levels. Often, a persistence may be combined with a scheduled probability to ensure a minimum duration for that probability and avoid multiple consecutive entries in the schedule.

Currently, only tasks, causal linkages, and effects in the SDT contain temporal information that is mapped to CAT's representation. Tasks, which are root nodes in the CAT model, include a scheduled probability that corresponds to the Task's Start Date attribute (measured as a C or D date). Tasks are given a persistence corresponding to the duration attribute. Effects and causal linkages are given a delay and persistence corresponding to the attributes of the same name specified in the "Timing" tab in the SDT Properties Editor. All other plan element types use the default delay and persistence (0 and 1 respectively). Only tasks feature scheduled probabilities. Users are free to edit temporal attributes of the translated plan directly in CAT.

3.3.3   Probability Mapping

This section describes the methods used to derive probabilities for CAT nodes and linkages during the translation of the SDT COA into a Bayes net.

Nodes

For nodes, CAT provides two main probability attributes: leak probabilities and scheduled probabilities (defined above). Leak probabilities represent a baseline probability of the node, assuming no active incoming causal linkages. When incoming causers or inhibitors become active, they perturb the node's probability upward or downward respectively from the leak value. Leak probabilities apply to all time-steps. Scheduled probabilities occur at specific time-steps with specific values. Scheduled probabilities override the leak and all incoming mechanisms during the specified time-steps.

While CAT allows scheduled probabilities on any node, the only plan elements in SDT that exhibit scheduled probabilities are tasks, at the root nodes of the causal model. They represent the initial trigger events that set the downstream effects and objectives and the rest of the plan in motion. The scheduled probability values are based on a discrete, qualitative scale shown in Table 2 below.

Table 2. Mapping of Task success probability labels to values

	Minimal	Low	Medium	High	Certain
Probability	0.05	0.25	0.5	0.75	1.0

Since the COA is evaluated in feed forward fashion, leak probabilities are set to 0 for all nodes. That is, their probability depends solely on the parent mechanisms, or, in the case of tasks, their scheduled probabilities.

Linkages

In CAT, linkages (formerly known as mechanisms) represent causal links between two nodes. Like nodes, they can have scheduled probabilities, but they do not have leak values, since a mechanism's probability is derived from that of its parent node. Linkages also feature two conditional probabilities that determine the parent action's effect on the child action:

- **H-probability** - This determines how a true parent node causally influences the child. Positive H values correspond to causers; negative values correspond to inhibitors. The absolute value of H is proportional to the degree to which the child action will tend toward true or false, given that the mechanism is true.

- **G-probability** - This determines how a false parent node causally influences the child. A positive G-value means that the child will tend toward true; negative means it will tend toward false.

When SDT translates the COA into CAT, causal linkage nodes between tasks and effects or between effects at successive levels are directly translated into a CAT linkage of the same name. Currently, all such linkages are causers, not inhibitors, since the feed forward translation words all tasks and effects positively from the blue point of view. The qualitative causal strength level entered in the Strength choice-box in the SDT Properties Editor determines the H-value. Table 3 lists the qualitative scale of causal strength labels and their corresponding H-values.

Table 3. Mapping of effect mechanism causal strength labels to CAT H-values

	Minimal	Weakly	Moderately	Strongly	Absolutely
Probability	0.05	0.25	0.5	0.75	1.0

All other CAT linkages created during the translation represent implicit links between SDT plan elements and their parents. For example, even though the SDT does not show a plan element connecting an effect to its parent objective, such a link is necessary in CAT in order to combine and summarize probabilities at progressively higher levels of the plan. The SDT encodes these levels of the plan in CAT as "Noisy ANDs," in which all parent nodes must be true for the child node to be guaranteed true, but the child has a small probability of becoming true if only a subset of the causers are true. For these cases, the H-values of the CAT linkages are inversely proportional to the number of parent nodes. For example, an objective that has three effects as children in the SDT will use an H-value of one-third for each linkage connecting each effect to the objective. This leads to a phased increase in probability of the objective as each of its effects becomes true. Finally, the SDT assigns an "AND probability" of 1.0 to the entire group of linkages, since attaining all effects implies the objective was also achieved. The same technique is used to assign causal strengths and group probabilities to linkages connecting objectives to the top-level COA node. Currently, the default delay and persistence of zero and one respectively are assigned to these linkages.

3.3.4   Assessment Results

The translation process described above is initiated when the user clicks the "Export COA for causal analysis" button in SDT's Analysis toolbar. The plan elements from the COA then appear in CAT's plan view and hierarchy view. CAT automatically arranges the actions and mechanisms hierarchically in a top-down layout to avoid overlap, with the COA node appearing at the top.

To begin an analysis of the plan in CAT, first save the plan using the "Save" option in the File menu. This generates output files necessary for CAT's Bayesian inference algorithms. Then, click on the "Build Bayes Net" icon in CAT's floating controls toolbar. A dialog appears prompting the user for the estimated plan duration and the number of simulation time steps. The first number is the estimated number of days spanned by the plan, as computed by the maximum sum of the delays and persistences over all bottom-up paths in the network. Adjust this number if a different time-span is desired. The second number is the number of actual time-steps CAT will use to sample probabilities in the plan over this time-span. It is recommended that users set the number of simulated time-steps equal to the plan length for simplicity. For plans with very large time-spans, divide the time-span by two or higher to improve the response time of the sampling algorithms through coarser-grained sampling.

To inspect probability results, choose the "Analyzer" view from the View menu. This adds a new Probability Profiles pane to the top of the Plan View showing an empty graph representing probability vs. time (see Figure 15). To graph the probability profile of any plan element in the COA, double-click on the CAT action in the plan view. The shape of the graph will be jagged at first due to noise in the sampling, but should converge to a stable estimate over time. Inspect the COA node for an estimate of the probability of COA success. The time slider from the "Time Selector" choice in the View menu may also be used to animate the probability coloring of the actions in the plan view and the probability values shown in the Hierarchy view (note that OAT does not support node probability colorings).
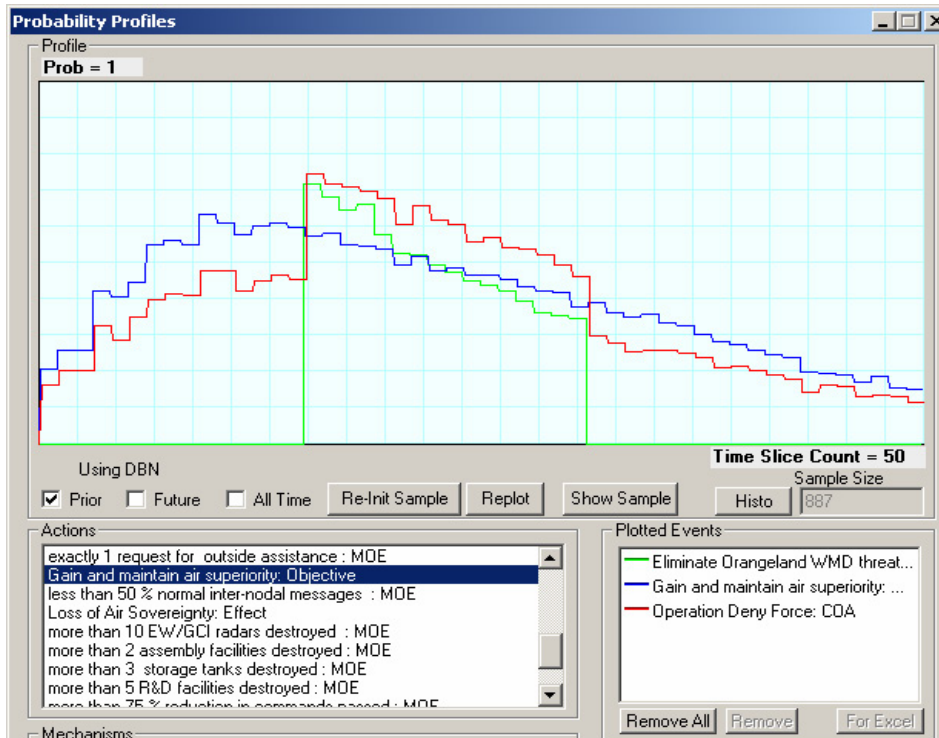
Figure 15. CAT Probability Profiles showing probability over time of COA success.

Figure 16 shows a montage of a simple SDT 1.1 plan fragment and corresponding CAT plan view and assessment view. Actual quantitative results are discussed in Section 4.1.1. As of SDT 1.4, the CAT tool was replaced by OAT, a hardened, well-documented version of CAT maintained by BAE Systems for purposes of the PBA/EBO JEFX '04 initiative and eventual transition to the operational community. The translation algorithm and user interface steps are the same.
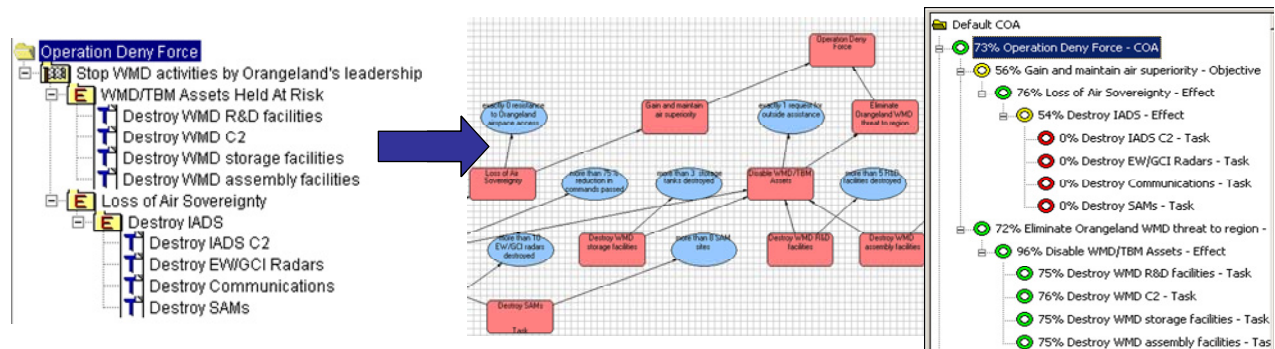
Figure 16. Resulting causal model and assessment view after exporting an SDT COA to CAT.

3.4 ENDSTATE INTEGRATION

While CAT was used to analyze the strategic-level SDT COA down to the effects and tasks, BAE's companion project Endstate, also developed under the EBO ATD, provided proof-of-

concept target system models and algorithms for analyzing the operational-to-tactical level COA in terms of its impact on enemy target systems. During the first year, a loose integration paradigm was used in which SDT and Endstate communicated solely through XML messages using an in-house XML router utility developed under DARPA Active Templates. Endstate initially provided its own rudimentary views for target system visualization and target system impact using the Matlab® quantitative analysis toolkit. Later, more comprehensive end-user oriented views were developed and integrated directly under the SDT application. The next three subsections describe these Endstate-related views, namely the Target Set Tool (Map View), Query Tool, and Undesired Effects Editor. The following two subsections describe the two main Endstate analysis capabilities, Target System Analysis and Option Generation, and how they were integrated with SDT. While Endstate broke new ground in model abstraction and cross-model integration, further research is needed to operationally validate and mature Endstate's target systems analysis and option generation algorithms.

3.4.1   Target Set Tool

The initial incarnation of the Target Set Tool was a Matlab window with a non-interactive blank background plotting enemy targets and linkages on an X-Y grid. This view was sufficient for testing the Endstate target system models and end-to-end data exchange with SDT for target system analysis during the first year. However, a more sophisticated map-based view was needed to interactively specify target lists and effectively convey Endstate's capabilities to end-users. During Year 2, the Endstate team performed two tasks to achieve a more robust platform for both demonstrations and future software extensions:

1. Conversion of the target system model data from flat files to a MySQL relational database

2. Migration of the Matlab-based target system view to BBN's OpenMap open source GIS tool.

The database migration effort was necessary both to support the sophisticated queries planned for the query tool (described in the next section) and to make the system more scalable as the number of targets in the system grows beyond what can be stored in memory. The migration to OpenMap not only made the Endstate demonstration more clear and compelling, but provided an integration hub for future geospatial features, target list development, and other SDT-Endstate interactions.
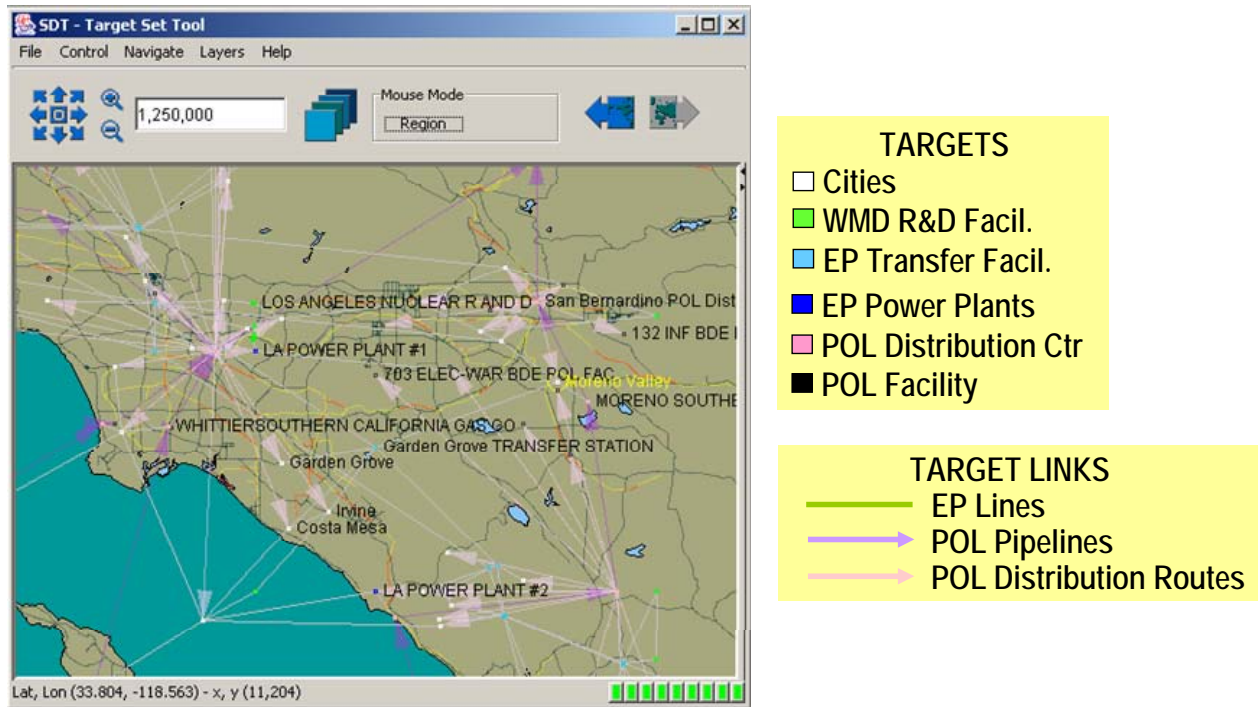
Figure 17. The Target Set Tool visualizes target system links and nodes.

Figure 17 shows a screenshot of the Target Set Tool. The main map pane is zoomed into the Los Angeles metropolitan area of the fictional enemy country Califon. The legend at the right of the figure (also available within the tool) shows the mapping of node and link colors to target type and link type. The user has turned on labels for certain targets (shown in black text) and selected one target (shown in yellow text). Other geographical features, such as roads, lakes, and political boundaries are displayed as distinct map layers that may be toggled on or off via OpenMap's Layers menu. These are derived from JEFX '00 ESRI shape files obtained from JFCOM. The top portion of the map window displays built-in navigation features for zooming and panning, as well as Endstate-specific features such as custom mouse modes for selecting regions and invoking tooltips on mouseover of nodes and links.

Another major function of the Target Set Tool is to create Candidate Target Lists (CTLs) for eventual incorporation into the SDT plan. The right side of the Target Set Tool initially was designed as two separate panes for *Direct Targets* and *Affected Targets.* This was later expanded to four alternative panes controlled via a combo-box, with new target lists for *Do Not Strike* and *Do Not Affect* (see Section 3.4.3). Direct targets are targets that we plan to strike against directly, and hence may be associated with a Task in the SDT COA. Affected targets are targets that we wish to affect, via one or more linkages to direct targets, and hence may be associated with an Effect in the COA. Users may add the currently selected targets in the map to one of these lists via a context-menu action. Targets are selected on the map by manually clicking on them, drawing a region around them, or selecting the results of an automated query (see next section). This may be repeated as necessary to incrementally add to the target lists. Finally, once the user is satisfied with a target list, the "Send to Plan Editor" button in the upper-right toolbar causes the list to automatically import into the current COA under the currently selected effect or task node in the SDT Tree View. The Affected Target list may only be imported under an Effect node, while the Direct Target list may only be imported under a Task node.

29

3.4.2 Query Tool

An important attribute of EBO is that it relies on both inter- and intra-dependencies of target sets, systems, and COGs for the purpose of planning for indirect and cascading effects. The Query Tool, developed in late 2002 in parallel with the Target Set Tool, was an attempt to facilitate such planning, providing an automated way to find and select related targets by following physical supply-and-demand linkages. It is launched via the Query Tool (question mark) button on the Target Set Tool. On the right-hand side it provides a series of tabbed panes denoting query restrictions of different categories. On the left-hand side it shows the current set of targets matching the currently applied restrictions. The system starts by showing all targets, up to some maximum number. As more restrictions are applied, the result set pane updates to display the smaller set of matching targets. Once the user is satisfied with the result set, the toolbar at the top allows for selecting them on the map, showing their names, or adding them to the Direct or Affected target lists. Figure 18 shows a screenshot of the initial version of the Query Tool delivered in SDT 1.2, in which the user is looking for WMD R&D targets within a given region. The bottom portion shows that the targets have been added to the Affected Target list, highlighted dark yellow on the map.
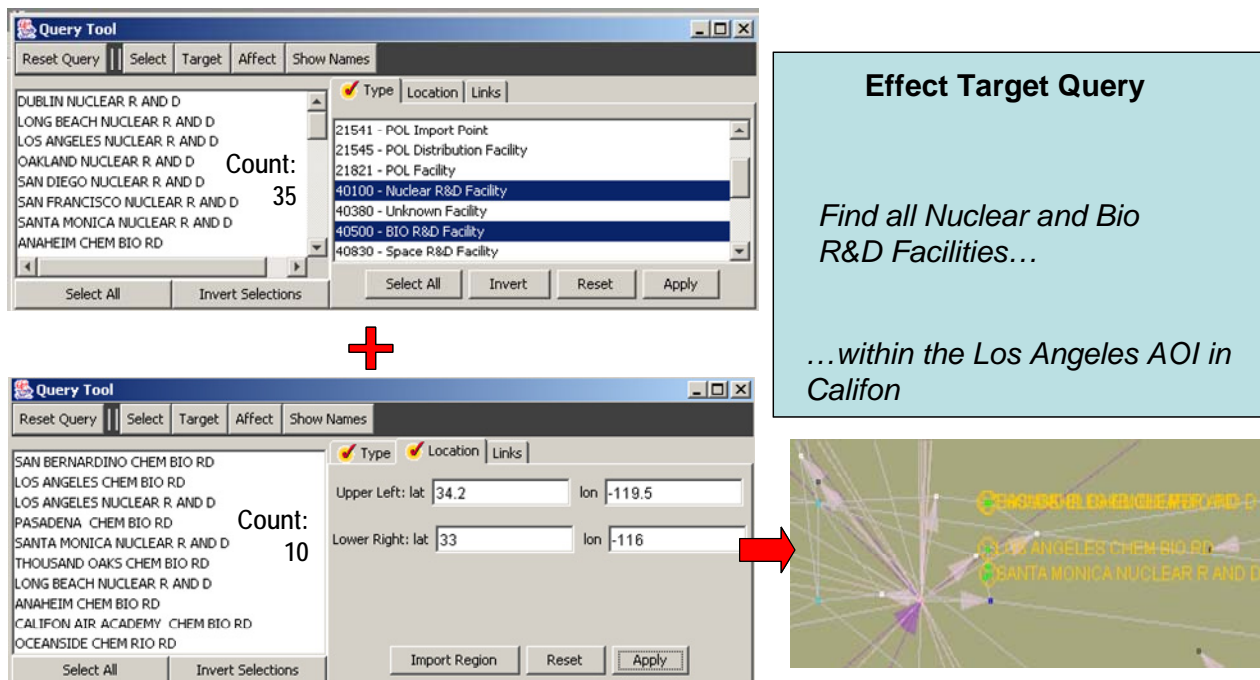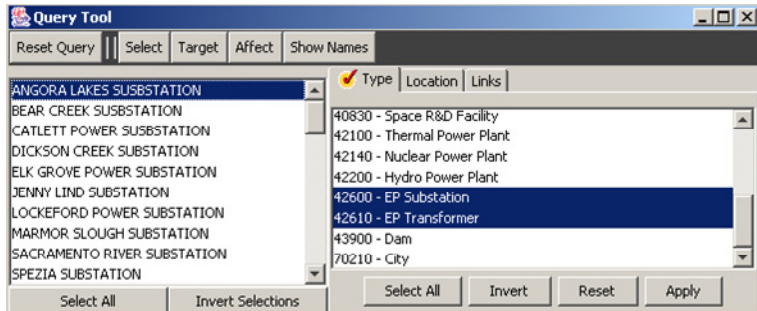
Figure 18. The query tool can be used to find targets to associate with indirect effects.
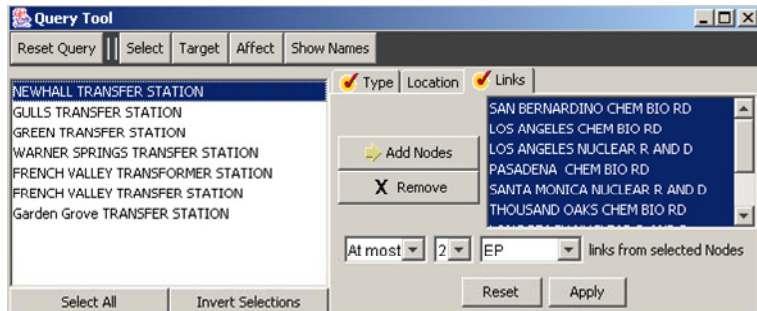
The first version of the query tool featured three restriction tabs:

- Type – A flat list of MIDB target category codes and category names (e.g. "Electric Power Substation")

- Location – Fields for specifying a bounding box by latitude-longitude corners, with a button fior importing the currently drawn region on the map

- Links – A more sophisticated restriction for identifying targets that trigger indirect or cascading effects. It must be initially seeded with a set of targets and finds all

connected targets within a specified number of links on a specified target system (Electric Power or Petroleum-Oil-Lubricants). This is useful for finding Direct Targets connected to a specified set of Affected Targets (see Figure 19). It may also be used to identify Affected Targets if the Direct Target set is known.

Figure 19. The query tool exploits target links to find strike targets to achieve desired effects.

In subsequent years we expanded the Type tab to provide a hierarchical chooser showing the complete set of MIDB category codes. These are five-digit codes with specificity increasing as the digits to the right are set. Users may select multiple category codes to look for conjunctions of target types. We also added three new tabs, shown in Figure 20:

- Gazetteer – Finds a target based on string match. All targets containing a substring matching the user's input will be shown.

- TSA – Restricts results to targets within one of six target systems: EP, POL, Fielded Forces, Air Defense, C4I and Telecommunications, and National C2

- BE – Allows the user to type in or cut and paste in a list of known BE numbers (unique target identifiers)
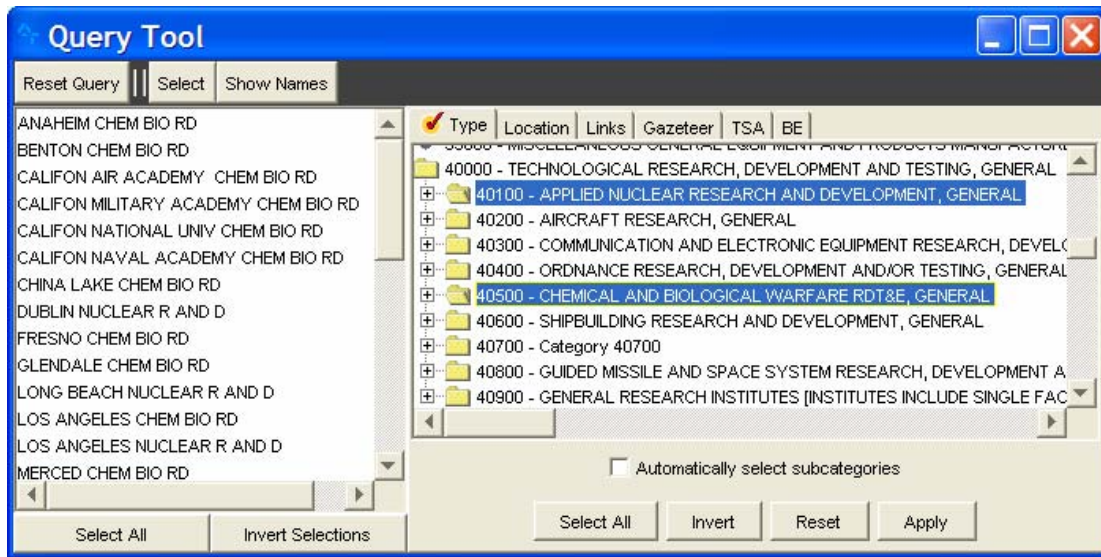
Figure 20. Hierarchical type restrictions and three new query tabs were added in SDT 1.4

3.4.3 Undesired Effects Editor

In late 2003, SDT began work on a new Tree View editor called the Undesired Effects editor, for the purpose of demonstrating an extension of Endstate's Target System Analysis capability to alert users when predicted target outage profiles incur pre-defined undesired effects. The algorithms and alert displays themselves were performed by the Endstate team. The SDT team focused on new editors for specification of No-Strike targets and targets with Effect Restrictions, which would later be included in the Endstate predictive analysis.

The new Undesired Effects editor was implemented within a new tabbed pane labeled Undesired Effects, appearing alongside a Plan tabbed pane containing the original Tree View. The new pane features a special tree view that overlaps with the main tree down to the level of COAs. Below that point, an entirely new set of plan element types are used. Under each COA, a user may add a single Do Not Strike (DNS) list (shown as a folder with a slashed 'S') and a single Do Not Affect (DNA) list (shown as a folder with a slashed 'A'). Under the DNS list, a user may add one or more target nodes, in the same way targets may be added under Task nodes in the regular Plan view. Under the DNA list, another level of information is required. Users must first specify one or more Restriction nodes. Each Restriction represents a condition (or effect) on a target or target set that we would avoid. For example, in Figure 21, the user has specified a restriction that electric power production should not fall below 90% of normal. Under a Restriction node, the user may enter one or more target nodes.

When a target systems analysis is performed within Endstate, SDT will alert the user if a target node property matching that specified in a DNA restriction matches the criteria specified in the restriction. These alerts appear as effect targets (targets under an effect node) highlighted light red in the Tree View. To see when and to what degree the undesired effect occurred, selecting the target node and clicking on the bar graph button in the Analysis toolbar brings up an outage profile graph. For DNA targets, the graph shows both the predicted outage profile in red alongside the desired minimum threshold in blue. When the red graph dips below the blue threshold, the DNA restriction has been violated. This capability was tested under somewhat

limited conditions, using the same ODF_CAOC_TARGETS demonstration scenario used for target systems analysis (see next subsection).
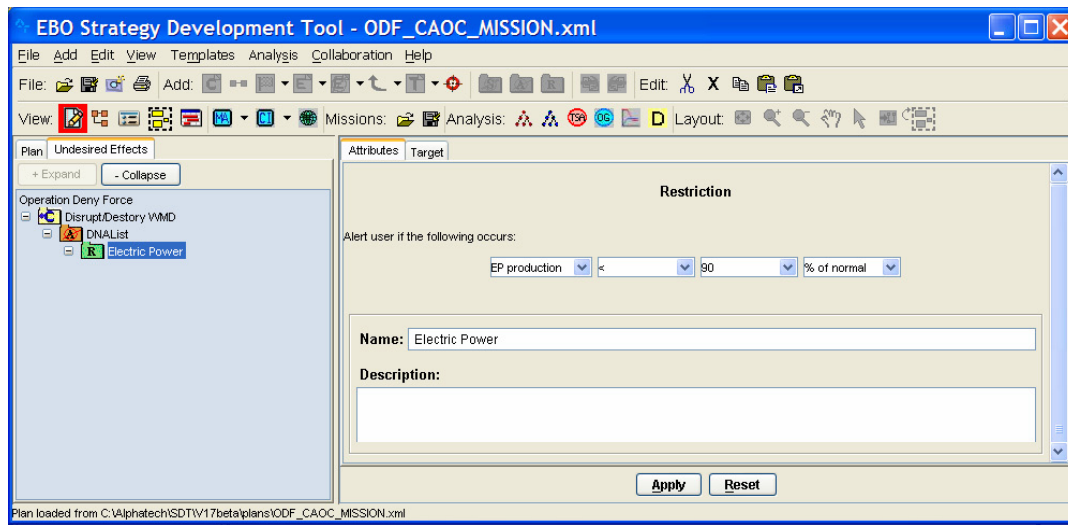
Figure 21. Endstate-level undesired effects can be expressed as restrictions on target sets.

3.4.4 Target Systems Analysis

During the first two years of SDT and Endstate, the two teams worked together over successive spirals to integrate Endstate's target systems analysis (TSA) algorithms. These algorithms simulate operational effects on enemy target systems based on cross-network models and optimization algorithms that predict worst-case enemy workarounds to blue-induced outages. Initially, Endstate used hard-coded target lists and a stub Matlab display showing the Electric Power and POL target networks in an X-Y plot simulating a map. With the completion of the Target Set Tool and Query Tool, BAE Systems had achieved a way for users to dynamically select targets, eliminating the need for hard-coded target lists. Before initiating TSA, a user must first create Direct and Affected Target lists in the Target Set Tool and import targets under at least one task and one effect (which must be downstream of a task with targets). Figure 22 shows the Direct and Affected Target Lists displayed in the Target Set Tool and SDT plan fragment from the final TSA demonstration (the plan named ODF_CAOC_TARGETS.xml). The Direct Targets are highlighted red and the Affected Targets are highlighted yellow.

Endstate requires two main types of information to perform a TSA:

- Direct and Affected Targets in the SDT COA, including references to parent nodes.

- Lowest-level Task and Effect plan nodes that contain targets.
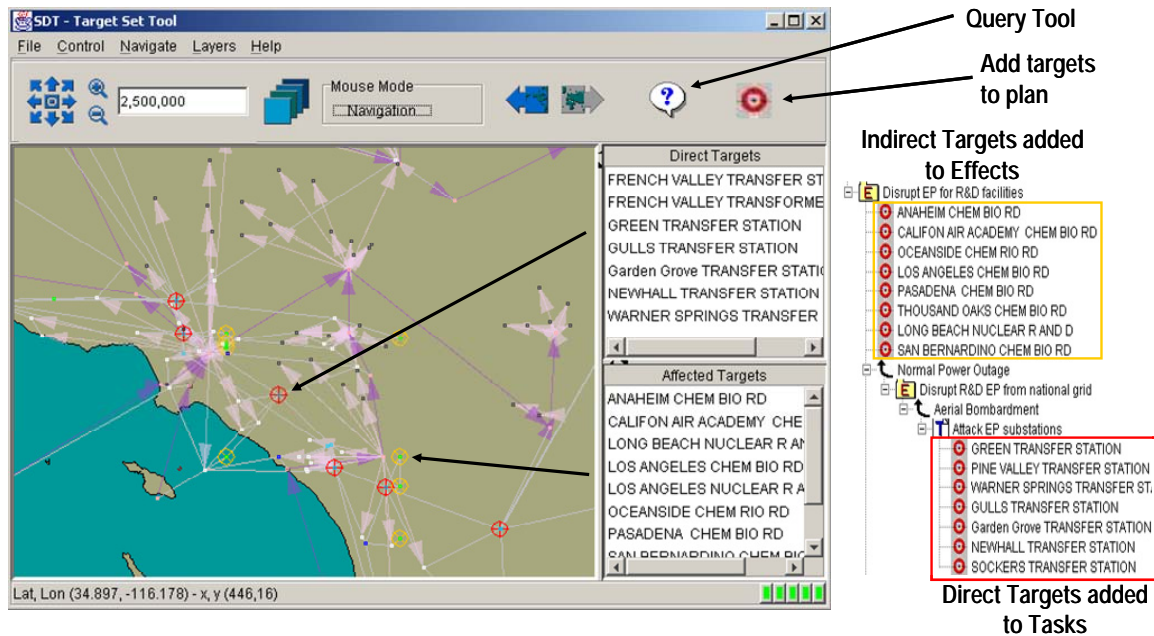
Figure 22. Prior to a workarounds analysis, the user must associate indirect targets with effects and direct targets with tasks.

Utilizing the XML Router utility developed under DARPA's Active Templates (White, 2004), we specified XML message formats for each of the above messages. SDT's PlanManager component (see Section 3.8) contains code for extracting all targets under tasks and effects and formatting them into a Target Set message in XML. It also walks the COA hierarchy and formats the lowest-level effect and task nodes encountered into a Plan XML message. Endstate receives these messages when the user clicks on the TSA button in SDT, through its XML Client which subscribes to Target Set and Plan messages. After the TSA algorithms are complete, Endstate sends an Augmentation XML message back to SDT's PlanManager, which has an XML Client subscribing to that message type. The Augmentation message contains a hierarchical series of plan node fragments combining the task and effect nodes with the target sets. Each effect node and effect target additionally contains an outage profile array specifying the EP or POL level over time at certain discrete time-steps in the plan (times when a strike occurs). The nodes for which outage profiles were reported are highlighted yellow in the SDT Tree View. The user can drilldown to see the outage profile graph by clicking on the bar graph button in the Analysis toolbar. In addition, the Target Set Tool (map) displays small bar graphs next to each target node showing the percent of normal level of EP or POL supply for that target. A time slider labeled with discrete time-steps used in the Endstate simulation can be used to step forward in time through the COA to see how the levels change over time. Typically they start at normal levels, then decrease at times when Direct targets are struck, then slowly rise to some equilibrium level if Endstate found a way for the enemy to workaround the outages. Figure 23 shows screenshots of the Endstate TSA map layer and time slider, and an individual outage profile graph for the Thousand Oaks Chem Bio R&D facility.
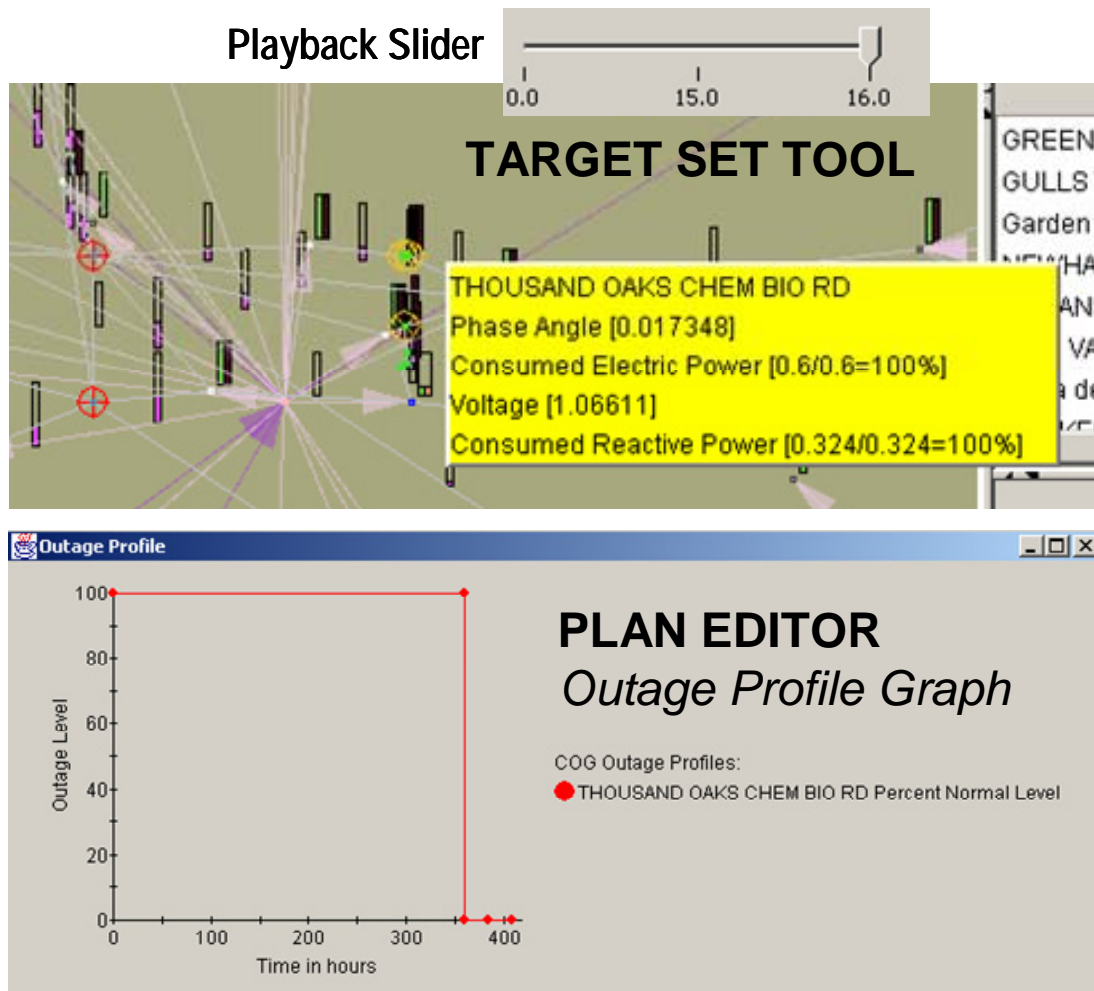
Figure 23. Workarounds prediction results are displayed as bar graphs next to each target, with further details shown via tooltips and outage profile graphs.

3.4.5 Option Generation

During Year 3, the Endstate team developed new algorithms for Option Generation. These algorithms require only Affected Targets (under effect nodes) and parent effect nodes as input, and produce as output a prioritized list of Direct Targets most likely to achieve the desired effect. These algorithms were designed to use whatever information is available. If both target system nodes and links are available, the algorithms use more sophisticated network flow algorithms such as min-cut max-flow to determine which strike targets will maximize the disruption of the specified Affect targets. If only nodes are available and no links, heuristic rules are used to look for high-capacity supply nodes in the vicinity of the Affect targets. These Option Generation algorithms are described in more detail in (Pioch et al, 2004).

To demonstrate the Option Generation algorithms, we created another variant of the Operation Deny Force SDT plan that retained the Affect targets but omitted the Direct targets. Figure 24 shows a screenshot of the relevant SDT plan fragment used as input.
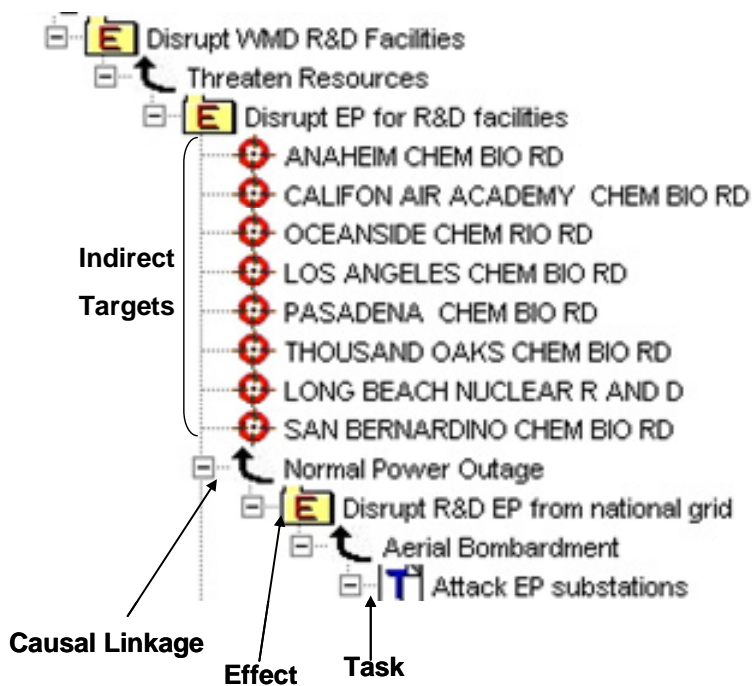
Figure 24. Option Generation requires only indirect targets and guidance on strike target types and methods.

SDT uses the same XML messages to initiate Option Generation when the user clicks on the OG button in the Analysis toolbar. The only difference is a mode field is set to "option generation" instead of "TSA." In this case, Endstate ignores the tasks and Direct targets and focuses on the effects and Affect targets.

When Endstate has completed its optimization, the results are displayed in a new tabbed pane in the Target Set Tool developed by the Endstate team. This pane shows the series of SDT plan fragments included in the original Plan message. Each fragment typically starts with Effect nodes which eventually lead to Task nodes under which Option Generation's suggested targets appear. The user may click on a proposed target to see it selected on the map display. Since targeteers would be unlikely to trust automated target selection algorithms sight unseen, the Endstate team included context menu actions for user approval or rejection of selected targets. These actions can be applied to an individual target or a selected set of multiple targets. When the user has finished approving or rejecting targets, they may choose the "Send targets to plan" button in the upper-right of the toolbar to import the approved targets into the plan. SDT knows where to import them because the plan fragments mirror nodes already contained in the plan. This process may be repeated for multiple task-effect fragments, or it may be performed once over all such fragments containing Affect targets. Figure 25 shows a screenshot of the Option Generation tabbed pane containing Endstate's suggested EP substation targets, with user-approved targets highlighted green.

After the user has finished importing Direct targets into the plan from Option Generation results, a TSA may be performed on the overall plan to verify that the chosen targets indeed achieve the desired effects.

Figure 25. After approving proposed strike targets, the user may add them to the plan.

3.5 COG ARTICULATOR FOR ADVERSARY MODELING

During year three (2003) BAE Systems built a new tool for SDT specifically aimed at adversary modeling, called the COG Articulator. This tool was a faithful implementation of the concept of the same name described in our original SDT proposal (Jones, 2000), after we discovered the limitations of building adversary models in CAT alone (see Section 3.6.3 for more in-depth discussion of lessons learned). The purpose of COG Articulator is to provide a freeform causal modeling environment for intelligence analysts and strategy planners alike to brainstorm about intra- and inter- COG dependencies and identify candidate interventions to bring about desired effects against enemy nodes of interest. While it retains the same core causal model representation as CAT, the terminology in some cases has been simplified to appeal to military end-users and new concepts such as red enemy vs. blue friendly nodes were added to support a wargaming-style modeling methodology. Also important, because it was implemented in Java as a constituent SDT tool, we were able to achieve tighter integration of COG Articulator with the SDT plan editor than would have been possible with CAT alone. Yet we did not lose any of our existing capability to interface with CAT because COG Articulator was built on top of the same Java abstraction layers that had already been in use in CAT. Thus, the current CAT view and DBN model would be guaranteed to be in synch with the user's COG model, allowing seamless alternation between modeling in COG Articulator and analysis in CAT (and later OAT).

The COG Articulator represents causal models using two main building blocks: COG nodes and COG links. These two types may be further subdivided into blue COA nodes, red COG nodes, observable (or indicator) nodes, causer links, and inhibitor links. Table 4 shows the icons associated with these different types of objects, which reflect their appearance in the COG model.

Table 4. COG Articulator model components.

Modeling Component	Icon
Red COG Node	⬭
Blue COA Node	⬭
Observable Node	⬭
Causer Link	↖+
Inhibitor Link	↖–

   The adversary modeling process is based on Step 3 of the Intelligence Preparation of the Battlefield (IPB) process, Evaluating the Adversary. The information on key Centers of Gravity obtained during IPB may be captured in a causal model in the COG Articulator. Typically, this is done by constructing a causal model consisting only of red COG nodes inter-connected by causers of varying strength. Then, during COA development, strategy planners examine the causal structure to determine critical nodes of vulnerability. These may be COG nodes on which an enemy is especially dependent, which, if disrupted, would prevent the enemy from achieving red goals or actions. Finally, based on this analysis, the planners may overlay blue COA interventions with inhibitor links against vulnerable red COG nodes and blue COA objective links inhibited by top-level red COG nodes. The effectiveness of the candidate interventions toward inhibiting red goals and achieving the blue objectives may be analyzed in OAT, as described in Section 3.6.

   Figure 26 shows an example of a COG model for the Operation Deny Force scenario built by BAE Systems for demonstration purposes. The model was originally built in CAT, then reauthored in COG Articulator. The top-level COG nodes pertain to the Leadership COG of a fictional enemy country Califon, who is threatening to invade a neighboring country Nevidah to take over its valuable mineral resources. Califon is developing Weapons of Mass Destruction (WMD) to help achieve its military objectives. One key blue objective shown in the model is to Prevent Use of WMD. To determine the probability of achieving this objective, we must first ask what is the probability the enemy will Deploy WMD, since this can be assumed to have the inverse probability (1-P) of our objective. Clausewitz theory teaches us that an enemy must be both Able and Willing to carry out an action, leading to the two supporting COG nodes Leadership Willing and Leadership Able. The former condition depends strongly on Support of Allies/Elites and Military Support, but depends only weakly on Civilians Content, as evidenced by the thin causal linkage, since this is an oligarchic dictatorship. The latter condition depends strongly on WMD Delivery Systems Operational, C2 Effective, and WMD Available. WMD Available in turn depends on a WMD "value chain" process of WMD Raw Materials, Electric Power, Transportation feeding Research and Development which in turn feeds Storage.

   Once we have elaborated the COG model down to the level of physical nodes we can attack or directly influence via military action or other spheres of action (e.g. DIME's diplomatic, informational, or economic spheres), planners can draw blue intervention nodes ('T' icons) that

directly inhibit these actionable red COG nodes. They can then analyze the COG model in OAT as a DBN to predict the eventual timing and degree of impact on the enemy's probability of achieving their goals (and inversely the probability of achieving related blue deterrence objectives). Based on the results, planners may adjust the timing, causal strengths, or choice of interventions until the specific desired effects are predicted to be achieved.
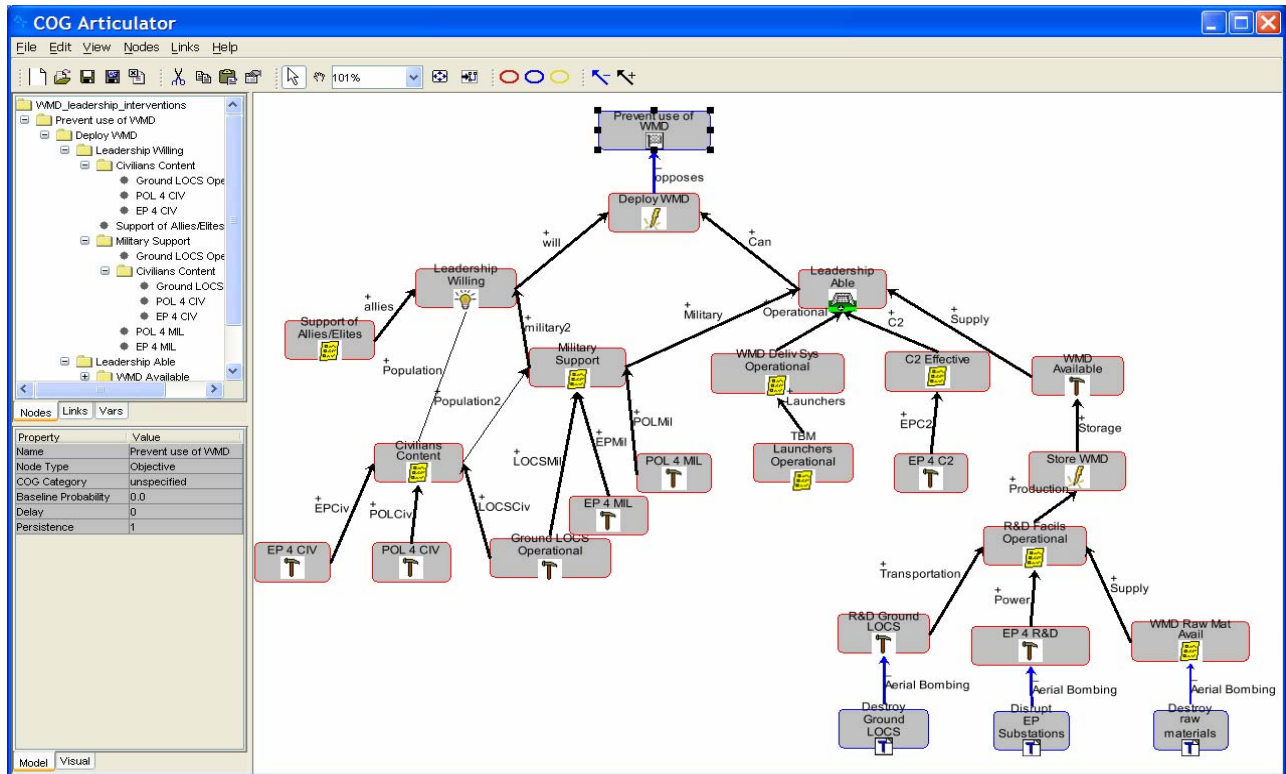
Figure 26. COG Articulator causal model for Califon WMD.

3.6 INTERVENTION ANALYSIS

While we had focused in year one on the feed forward method of blue COA analysis described in Section 3.3, our original intention had been to use CAT for building and analyzing adversary (red) COG models. Blue interventions could then be introduced into these models to predict the impact over time on enemy COGs. This constitutes a more effects-based approach in which the affected enemy nodes and causal linkages are made explicit. Moreover, the mechanism for achieveing high-level effects is also made explicit by the chain of enemy nodes and causal linkages between the blue intervention (action) and the object of effect. Interior red nodes not explicitly mentioned in the blue COA become candidates for indicators or measures of effectiveness (MOEs) in the blue COA. Our approach, while more specific in that it calls for temporal causal models, is consistent with the ONA methodology espoused by JFCOM in the Theatre Effects-based Operations program. The following sections describe early examples of COG models built directly in CAT, a concept for extending those COG models via operational-level augmentations from Endstate, methods for analyzing interventions using COG Articulator and OAT, and how to automatically generate COA fragments from the analysis.

3.6.1 Intervention Analysis in CAT
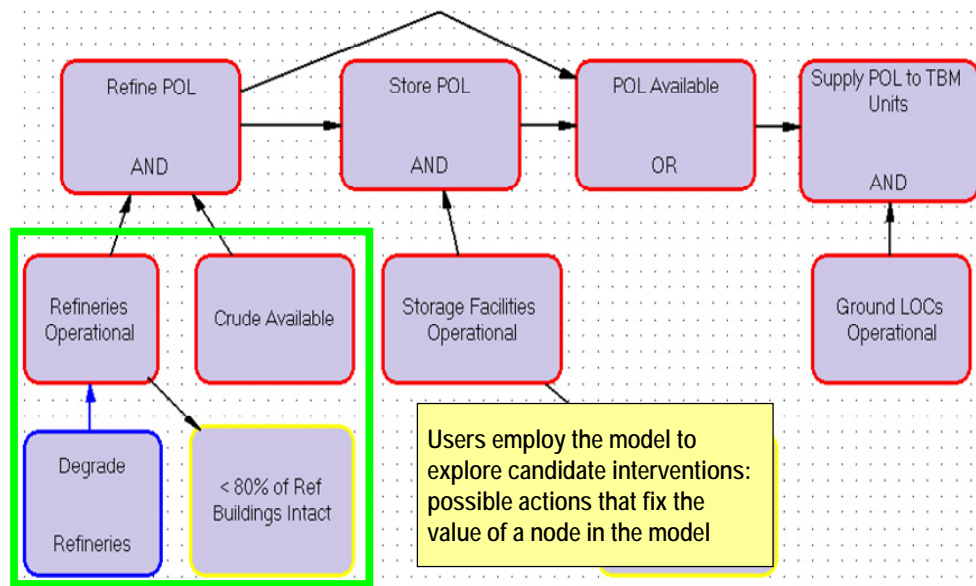
Figure 27. Early concept of a COG model for POL with candidate blue intervention.

During year one, in parallel with the direct blue COA analysis described in Section 3.3, we developed small-scale examples of strategic COG models using CAT. These models intentionally served as abstractions of the primary Endstate target systems, Electric Power and POL, in order to convey how Endstate analysis results at the operational level might inform and extend the CAT strategic COG models. Figure 27 shows an example of a simple COG model depicting the POL "value chain," illustrating causal dependencies among the key steps of refining, storing, and supplying POL. The baseline COG model contains only the red nodes shown in the figure. Note the "AND" and "OR" labels at the bottom of these nodes, which convey whether the supporting causes contribute conjunctively or disjunctively to the probability of the common child node. For example, Refine POL requires both Refineries Operational and Crude Available to be true, while POL Available may be true if either Refine POL or Store POL is true. Running CAT on the baseline model reveals the "normal case" probabilities of these red nodes in the absence of blue interventions. Because of the delays inherent in the nodes and linkages, the further downstream the node in the process, the later it reaches its steady-state probability, but, assuming there are no scheduled probabilities and no feedback loops, eventually the entire system will reach an equilibrium state. The steady state of this baseline model features a high probability of enemy success in the goal node Supply POL to TBM Units.

Given that we, the blue side, wish to reduce this probability as much as possible, intervention analysis begins with identification of candidate COG nodes that 1) are causally upstream of the node we wish to affect, and 2) nodes against which we have the power to intervene directly. In this example, a candidate blue intervention Degrade Refineries is shown to be inhibiting the Refineries Operational node via the blue causal link. If we schedule this intervention to occur at a particular time, for a specified duration, with an estimated probability of successful *task performance,* we can use CAT to predict when, how long, and to what degree the impact will be felt on the object of effect Supply POL to TBM Units. Intervention analysis involves iteratively posing candidate interventions and analyzing their downstream impact on the enemy, refining the timing and combinations of interventions along the way until the desired effect is achieved.
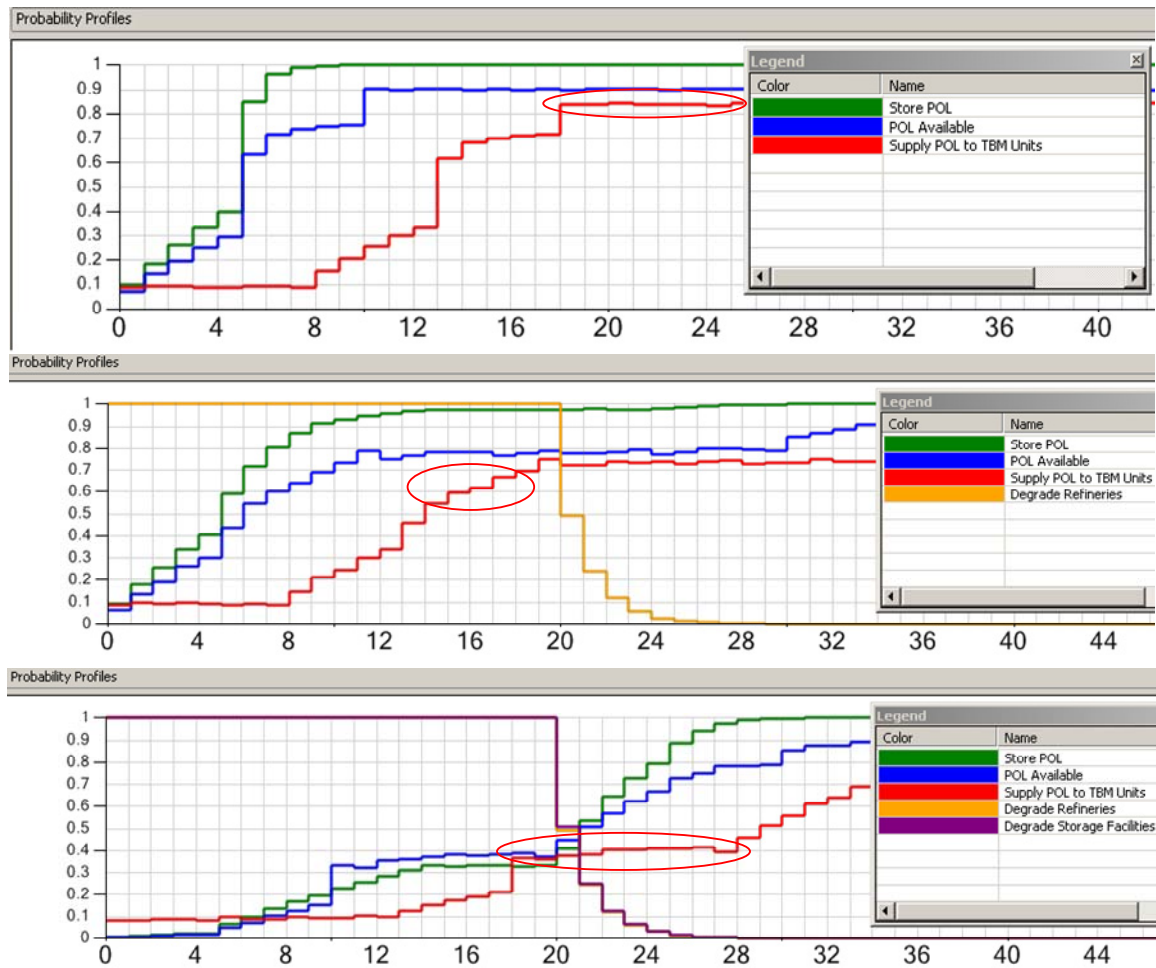
Figure 28. CAT's probability profiles reveal the most effective combinations of interventions.

For example, Figure 28 shows the predicted probability profiles for the nodes Store POL, POL Available, and Supply POL to TBM Units under three different blue intervention scenarios. The first display results from the baseline COG model with no interventions. This model shows that the Supply POL to TBM Units goal node (red line) reaches a near-peak of 0.6 at Day 13 and a steady-state peak of 0.7 at Day 18. The second display shows the impact of a single intervention, Degrade Refineries, scheduled at Day 0 with a persistence of 20 days, as shown by the yellow line. Because POL Available can be true either if Refine POL is true or Store POL is true, this intervention had an almost negligible impact on the goal node. However, when we add a second intervention, Degrade Storage (purple line), we achieve a significant reduction in the goal node's probability, peaking at 0.4. We also see that due to delays and persistences inherent in the red model, this effect lasts until Day 28, when the goal probability starts climbing back to its original steady state.

Finally, note the yellow node in Figure 27, "< 80 Refinery Buildings Intact." This is an example of a MOE stated positively from the blue side's point of view, thus it would have been added to the model in combination with the sample blue intervention. MOEs do not play a significant part in intervention analysis to support COA planning, but are critical during

41

execution in that they help confirm or disconfirm achievement of desired or intermediate effects. Both CAT and OAT include special evidence sampling algorithms that account for "soft evidence," which are observed probabilities at particular times on MOE or indicator nodes. These algorithms update the probability profiles for the rest of the nodes both upstream and downstream of the observed MOE node to account for the evidence. Typically, the further away the node topologically from the MOE and the further in time the less the impact of the evidence will be felt. This temporal evidence inference capability aids the campaign assessment process by revealing effects not achieved due to failed actions or suspect causal linkages or intermediate COG nodes. It can also highlight unexpected positive results, such as an effect predicted to be achieved earlier than expected, allowing friendly assets to be reallocated to other important tasks.

3.6.2   COG Model Augmentations from Endstate

The temporal causal model representation used in the COG modeling approach described in the previous section are especially suited to strategic-level adversary modeling, for which there is often limited "ground truth" beyond the mental models or abstract and often sparse written studies of ISR analysts or strategy planners. The representation supports rapid brainstorming and externalization of these mental models to foster a consensus among the planners on the enemy's COG and cross-COG dependencies that are relevant to their strategic objectives. On the other hand, the Endstate project deals primarily with hard target systems embodied in a network of physical nodes and links, which can often be determined through ISR sensors and in-country intelligence. Endstate's physical networks and workarounds algorithms represent operational COG models that complement the strategic COG models developed in COG Articulator or OAT. An original joint objective of SDT and Endstate was to develop automated algorithms for *model augmentation,* that is, extending the manually-authored strategic COG model with a fragment of an Endstate operational COG model, translated into a Bayesian network structure. This augmentation would be attached via causal linkages to an existing red COG node semantically equivalent to a high-value target in Endstate or an abstraction of some subset of an Endstate target system. It would itself serve as an abstraction for the more complex physical network model employed during TSA, typically containing only those target system nodes on the network on paths connecting direct targets to affected targets. This allows planners to perform intervention analysis at a higher level of fidelity than with the original strategic COG model alone.

Figure 29 shows a notional example of an augmentation for the familiar POL value chain model. The red node Ground LOCs Operational has been expanded with a DBN fragment summarizing the specific routes and chokepoints the enemy could use to transport POL to TBM Units, with the causal linkages and conditional probabilities encoding the AND/OR dependencies among these infrastructure nodes. For example, the enemy can transport POL via two alternative avenues of approach, AA1 or AA2, thus the Ground LOCs Operational node encodes an OR relationship on these conditions. But AA2 requires both Mobility Corridor 2A AND Mobility Corridor 2B to be operational. Finally, the new leaf COG nodes in the augmentation represent specific bridges along these corridors that may be targeted as chokepoints to prevent POL Supply from reaching TBM units. By extending the original strategic COG model down to the level of relevant target system nodes, planners may discern candidate tactical targets to associate with tasks and effects in the COA. They may analyze the interventions in OAT for rapid validation and "what-if" prediction process, or they may use the

more detailed workarounds optimization in Endstate for more rigorous validation of target selection.
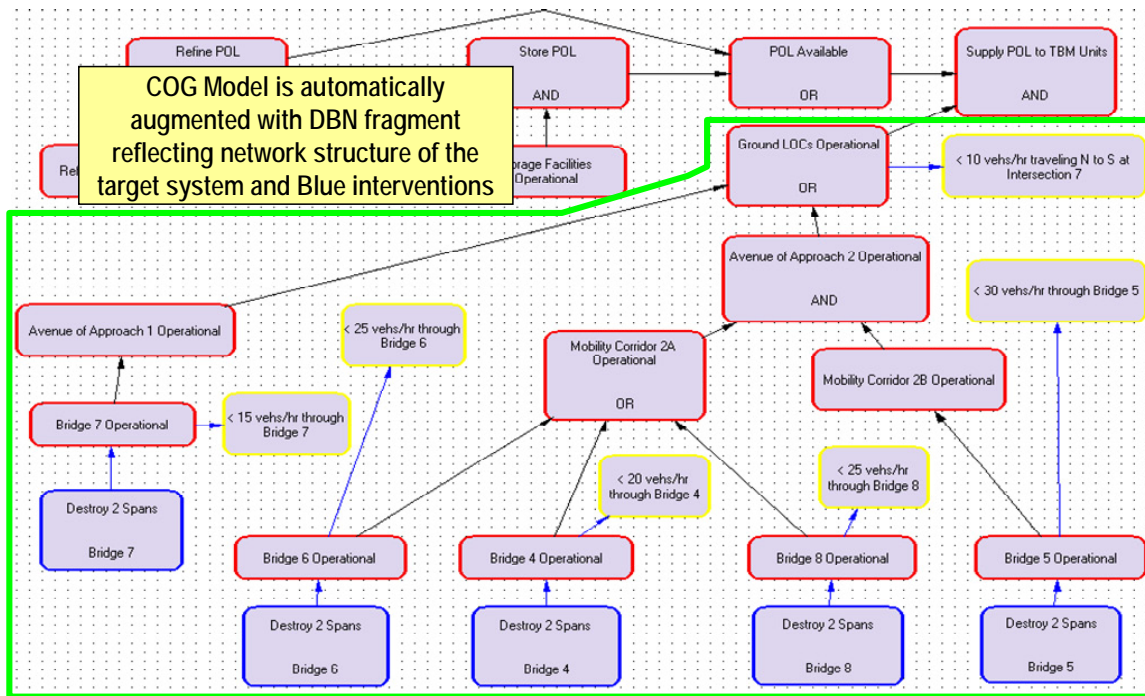
Figure 29. Concept for DBN augmentation summarizing the relevant portion of Endstate's target system network structure.

While these initial hand-coded examples seemed promising, after over a year of consideration and discussion with the Endstate team, we realized that achieving an automated augmentation capability would be extremely difficult to implement accurately and might be only marginally useful, perhaps even misleading in many cases. The primary reasons are twofold:

1. Endstate uses a dynamic closed-loop optimization algorithm based on economic supply and demand theory and network flow of commodities. When an event occurs, its direct effects are propagated in feed-forward fashion, and then the enemy dynamically reacts to compensate for those effects. OAT, on the other hand, uses Bayesian networks, which are inherently open-loop and limited to feed-forward predictive analysis (aside from evidence inference). The original CAT included a special "feedback mechanism" linkage, but its applicability to abstraction of conventional optimization techniques like linear programming were not clear.

2. The primary outputs of Endstate are *outage profiles,* which predict time series of actual values of continuous variables such as EP supply. In contrast, the outputs of OAT and CAT are *probability profiles* over Boolean variables that have only two possible states, true or false. The semantics of these two types of models are inherently inconsistent; mapping a continuous Endstate variable to a probability would yield increasingly suspect results as the Noisy-OR formulas for synergistic causes diverge from the additive network flow formulas for multiple supply sources. We must either resort to "watering down" the continuous outage profiles by comparing to an arbitrary threshold (true if above, false if below), or we must extend

43

CAT to handle multi-valued states that more finely approximate the continuous range of values produced by Endstate. The former idea risks masking potentially important effects because they happen to fall under the arbitrary threshold. We investigated the latter idea, but eventually realized that a non-Boolean CAT would require a completely new and much more complicated causal model abstraction approach, throwing out the simple influence-based model that made CAT attractive in the first place.

Despite these misgivings, there are likely special cases or classes of cases where at least the immediate feed-forward effect prediction in Endstate could be reasonably approximated by a Bayesian network. One case in point is the LOCS example in Figure 29. Because the flow of POL along LOCS is always feed forward, and road networks lend themselves to a Boolean interpretation (a node is either open or closed), the flow of probability through the DBN might reasonably approximate the output of Endstate's LOCS Workarounds Reasoner (a stand-alone Endstate tool included in year one). One method for approximating an enemy's dynamic workaround (e.g. bridge repair) that avoids the need for a reactive close-loop component is to encode the workaround time as a persistence limit on the blue inhibitor. Alternatively, the system could introduce a new overriding red node "Bridge Repaired" that kicks on at the time Endstate first predicts the workaround to succeed.

Further research on model abstraction is needed before automated model augmentation can be viably used operationally.

3.6.3 Analyzing COG Articulator models in OAT

During year two, we attempted to integrate SDT with CAT more tightly to support the above intervention analysis process. Each task and effect node in SDT included an "Affected COG Node" property field. This could be filled in with the name of a red COG node in the current CAT model. An "import red model" button in the Analysis toolbar could be used to query CAT for the names of all nodes in the model, such that they would be made available in a pull-down menu within this property. Then, after making all desired associations, the user could click another "Overlay blue COA" button in the Analysis toolbar to automatically export all COA task nodes to CAT as blue interventions inhibiting their associated COG node. Blue effect nodes were not exported but were assumed equivalent to the inverse probability of their associated effect nodes (e.g. "disrupt POL refineries" would have the inverse probability of the red COG node "Refine POL"). This was demonstrated during the April 2002 EBO PMR. This implementation of the intervention analysis process was unwieldy, and required many manual steps just to verify in CAT what had already been planned in great detail in SDT. We came to realize that intervention analysis can be more rapidly performed in a freeform environment like CAT, and that the elaboration of the blue tasks and resulting effects is more appropriate to perform in SDT only *after* validation of the choice of interventions in CAT. However, CAT was designed to be a general tool for causal modeling and lacked any explicit support for adversary modeling or blue intervention representation. Another practical consideration was the difficulty of extending CAT's API and tightly integrating across the Sun Java and Microsoft C#/C++ programming languages. While the AFRL CAT team was very responsive to BAE Systems' requests for new features and API methods, we decided that developing a separate Java-based adversary and intervention modeling tool would allow the SDT team more flexibility to integrate seamlessly with the Plan Editor and would provide an effective bridge to intervention analysis in

CAT. Indeed, all these benefits were realized prior to JEFX '04, contributing to a qualified endorsement of the methodology from users with modeling experience.

With the event of the first COG Articulator prototype, we had thus arrived at a more refined concept of operations for intervention analysis in which lightweight model authoring occurs in COG Articulator, while quantitative probability analysis occurs in OAT. Because the COG Articulator was developed in Java as a tool in the SDT suite, blue COA nodes, in particular interventions, could more easily be exported into the COG model and used as candidate interventions. Alternatively, users could specify interventions directly in the COG model via the COA Node toolbar button. Because COG Articulator was built on top of the same DBN interface layer used previously with CAT, it was possible to enable a "live" connection to CAT such that all modeling activity is dynamically mirrored in CAT. This allows users to seamlessly switch back and forth between COG modeling and CAT analysis without extra steps of importing/exporting models.
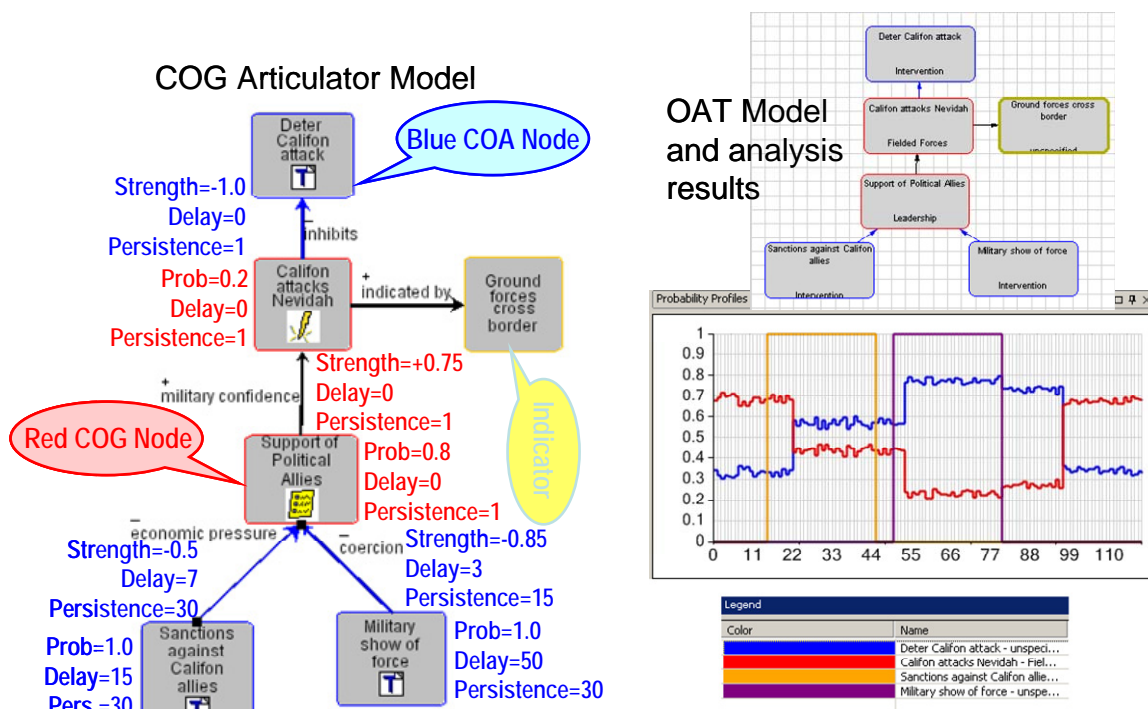
Figure 30. COG Articulator is tightly integrated with OAT for on-the-fly predictive analysis.

Figure 30 shows side-by-side screenshots of a simple COG model with interventions as it appears in COG Articulator on the left, and OAT on the right (as of SDT 1.4, SDT switched from CAT to OAT to better support JEFX '04). Most of the core CAT DBN parameters are retained in the COG Articulator, but some have been renamed to target a more operational user base, for example, "Strength" is used in place of "H-value." The names and colors, even the layout, of nodes in OAT are equivalent to that of COG Articulator, making it easy to understand probability results for specific model nodes and determine where to make model changes to improve results.

The next section describes how planners may use COG Articulator to close the loop between intervention analysis and strategy.

3.6.4 Incorporating causal chains into the blue plan

Based on the lessons learned from our earlier attempts at tight intervention analysis using SDT and CAT, we designed COG Articulator with an interface feature for exporting causal chains stemming from blue interventions back to SDT in the form of blue COA fragments. The assumption is that after the planning team has determined an appropriate set of interventions validated by favorable probability results in OAT, they will want to make these interventions and the subsequent effects on the enemy part of the their COA. Since the model itself contains an explicit representation of these interventions and effects as blue-to-red causal chains, it would be possible to implement an algorithm for converting the nodes and linkages in the causal chains to their corresponding SDT plan node equivalents.
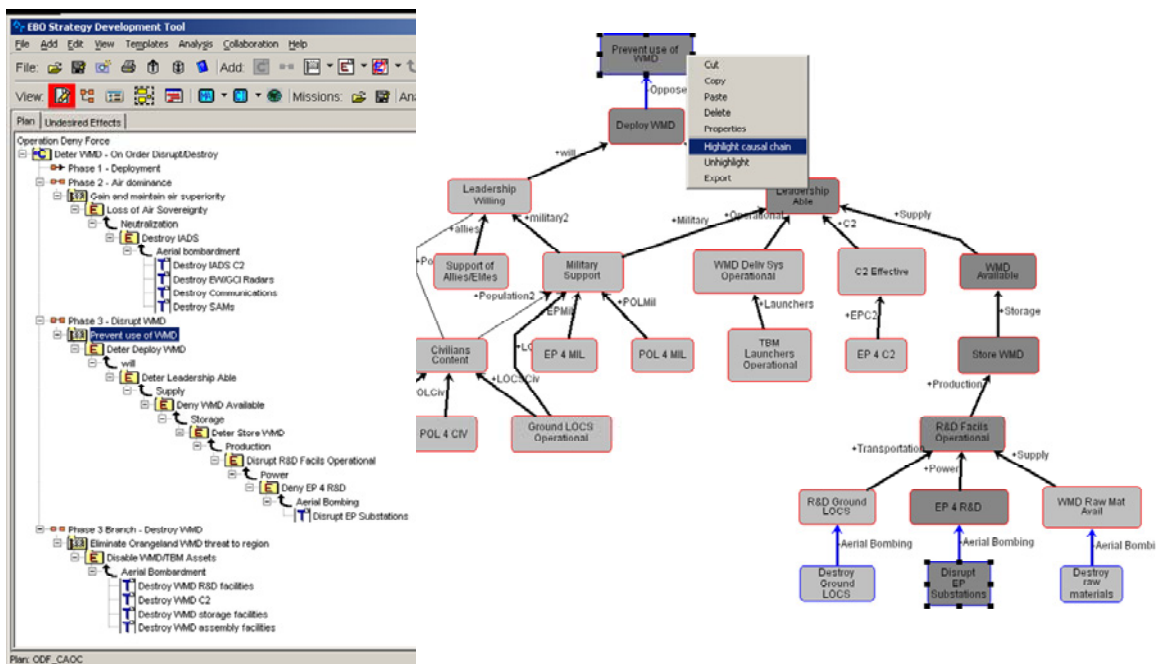
Figure 31. The causal chain of effects resulting from the best candidate intervention from comparative analysis in OAT may be imported into the Plan Editor as a blue COA fragment.

The process, illustrated in the screenshot shown in Figure 31 requires three main steps:

1. Select the SDT node in the Tree View under which to import the new plan fragment.

2. In COG Articulator, control-select the two endpoints of the causal chain. Right-click on one of them and choose Highlight Causal Chain.

3. Right-click again and choose "Export." A blue COA fragment equivalent to the highlighted chain appears under the selected SDT node in the Tree View.

The algorithm itself is implemented in the SDT PlanManager component, which acts as mediator between COG Articulator and the SDT Plan Editor. The algorithm first finds all

connecting paths between the two endpoints by walking the graph in depth-first fashion from both endpoints, saving all successful paths. Then it walks through each path, converting COG Articulator objects to SDT objects using essentially the inverse of the blue COA translation rules described in Section 3.3. These rules are summarized in Table 5 below. After translating each object, the algorithm walks the paths once more to set up the equivalent parent-child relationships among objects in the SDT plan fragment. The root node of the resulting plan fragment is then handled similarly to mission template fragments in that its type must be a legal child type of the currently selected node, otherwise an error message is displayed.

Table 5. COG Articulator translation rules for exporting intervention chains to SDT.

COG object	COG type	SDT object	Comments
Blue COA node	Objective	Objective	Same name.
	Intervention	Task	Same name. Success probability, start time and duration derived from scheduled probability and persistence.
Red COG node	Action or Goal	Deter Effect	Verb prepended to COG node name, to state as blue desired effect. Delay and persistence are preserved.
	Belief	Convince Effect	"
	Resource	Deny Effect	"
	Condition or Generic	Disrupt Effect	"
COG Link	Causer	Causal linkage	Causal strength label derived from link strength. Delay and persistence are preserved. Linkages connected to objectives have no explicit equivalent in SDT, due to direct objective/effect relation.
	Inhibitor	Causal linkage	Causal strength label derived from link strength. Delay and persistence are preserved. Now considered a causer.

This capability effectively closes the loop between effects-based modeling, analysis, and planning.

3.7   ENHANCEMENTS FOR JEFX '04

Beginning in April 2003, BAE assisted AFRL on a proposal for a new PBA/EBO initiative for JEFX '04 involving the primary EBO tools described in this report, SDT and CAT (or OAT as it turned out), with Endstate included as a component of SDT. Later that summer, word came back that the initiative was accepted. With this announcement, the focus of the project shifted from research to transition. In addition to the base Year 3 ATD funding of $289K, the project received in its last two active years $200K from ESC, $400K supplementary funding from JEFX, and most importantly over $720K from AFRL to ensure that development of new strategy capabilities for the warfighter could occur in parallel with the costly demands of technical support for JEFX '04. The AFRL discretionary funding was critical in enabling the team to be responsive to end-user feedback, developing major new capabilities between spirals in addition to enhancements to existing components. While the final state of enhancements made for JEFX to original views is reflected in the earlier sections, this section covers the new capabilities that were developed to varying degrees to meet the significant demands of the 8th AF, who served as the JEFX '04 players.

3.7.1 Plan Editor Redesign

The first major task, as early as August 2003, was to replace the Plan Editor's grammar-driven NL UI templates with more conventional UI widget design. This was motivated by several considerations:

1. The space of statements that could be created in the NL UI grammar were not sufficiently complex to warrant the number of clicks and phrases required to complete a statement.

2. While the NL UI enabled us to rapidly experiment with different statement grammars during the research stage, the complex Java/Prolog implementation was not well-suited to maintaining a centralized Java object data model, even after adding semantics. A Model/View/Controller architecture for the Plan Editor was deemed critical to support rapid development for JEFX '04.

3. We anticipated a need to simplify the Plan Editor user interface as much as possible to satisfy the JEFX '04 time-strapped end-users.

Thus, we took the opportunity not only to migrate from the NL UI representation to a conventional UI, but also to redesign the entire suite of plan node editor screens to more thoroughly cover the information embodied in the JAEP process described in JP 3-30 (2003). Our experience working with end-users and SMEs at the EBO workout also partly inspired the new screen designs. We prepared and peer reviewed detailed mockups of each new editor in PowerPoint, then developed a UI library of reusable widget groups, such as label/textfield pairs. Our new Plan Editor implementation also utilized the Model/View/Controller pattern at multiple levels, applying it both to individual fields and to entire object-level updates. To maintain compatibility with the ISX data representation and existing views (Objective and Timeline views, still present up to SDT 1.4), we decoupled the controller from the concrete model implementation via an interface package. This allowed us to extend the ISX plan ontology classes at will in our own subclasses, make use of the extended data in our own editor screens, while the objects still appeared to the original ISX views as the expected ISX classes. When the ISX editors were later replaced with enhanced BAE views, the impact on the rest of the architecture (model and controller) was minimal. In addition, this new, simplified and more modular Plan Editor UI framework served us well in the months ahead as we continued to refine both the user views and data content in the Plan Editor.

3.7.2 Collaborative Planning

One of the three main focus areas of JEFX '04 was Network-Centric Infrastructure, which includes tools to facilitate collaboration among warfighters. Thus, in early 2003, AFRL asked us to come up with options for extending SDT to support collaborative planning. We proposed the following three options, in increasing order of scope:

1. Manual publish and insertion of plan fragments using strategy web services (XML pub/sub, tied to backend DB)

2. Message-based tasking for remote elaboration of plan fragments

3. Fine-grained collaboration for change notification, database persistence, and real-time updates

Option 1 was a minor extension of SDT's existing mission templates capability, in which the templates could be saved to the Strategy Web Services for later importation by other planners. Option 2 involved new messaging tools to enable high-level planners to delegate the elaboration of an objective to a planning specialist, helping them track the workflow status of each delegated task. Option 3, by far the most ambitious, focused on migrating from a file-based to database persistence mechanism for plans and automatic notifications over the network to remote users to keep everyone's view of the plan synchronized. The cost estimates for these options were low, moderate, and high respectively, ranging from a few person-weeks to nine man-months. Based on these estimates and the future JEFX funding expected at the time, AFRL recommended Option 2. The next subsection describes our subsequent implementation of message-based collaboration. Based on feedback received during JEFX '04 Spiral 2 (see Section 4.3.2) in which users expressed a desire for Option 3 capabilities, AFRL approved work on Option 3, which is described in the subsequent subsection.

Message-Based Collaboration

SDT is primarily aimed at members of the Strategy Plans team within the AOC Strategy Division. This team includes a chief of strategy and a variety of planning specialists knowledgeable in different aspects of air operations such as information warfare, intelligence analysis, logistics, and more. No single person develops blue strategy within this team, rather, they tend to work in pre-assigned committees that later come together to discuss the pros and cons of each approach. JAOC doctrine requires that multiple blue courses of action (COAs) be developed, so these committees often work in parallel on alternate strategies addressing different top-level guidance. Thus, we designed our collaboration tools to support a top-down workflow in which high-level team officials such as the strategy chief delegate COAs, phases, or objectives to one or more planning specialists. Each specialist may actually be the "scribe" of a committee working together on the assigned problem. When a specialist has completed a plan fragment, it may be remotely submitted to the original requester for review. This process may be applied recursively at successive levels of planning, or in parallel with simultaneous delegations at the same level.

Our system design for this style of collaboration extends mission templates' file-based approach by providing 1) a flexible messaging architecture and 2) collaboration tools for assigning and tracking planning tasks among small teams, described in the following subsections.

Because SDT is a pure Java application, we based our design on established Java 2 Enterprise Edition (J2EE) standards and design patterns. The design features a three-tier architecture:

- **Bottom tier**: Apache's JBOSS application server is used for automated message routing and queuing via the Java Messaging Service (JMS).

- **Middle tier**: A *collaboration server* was implemented to manage user accounts and messages and log all user messages and planning tasks. The collaboration server uses JMS to communicate with the application server.

- *Application tier*: A suite of graphical collaboration tools for sending and viewing messages and managing planning tasks are tightly integrated with existing SDT plan authoring tools.

Building our solution on JMS eliminated the need to implement a generic message framework, allowing us to focus most of our efforts on the application-specific functionality. The collaboration server in the middle tier is responsible for persistence of user accounts and user message queues, enabling a planning team to pick up wherever they left off after system shutdown and restart. The application tier uses a custom Application Programmer Interface (API) to communicate with the collaboration server. This API abstracts the details of the JMS messaging architecture in the underlying implementation, allowing us to easily migrate to alternative communication technologies in the future, such as Simple Object Access Protocol (SOAP).

Three types of messages are supported in the application tier: *text, collaboration*, and *system messages*. Text messages are similar to those used in on-line chat tools or simple email applications. A user simply chooses the recipient from a list of known usernames and types in a message in a text field. After it is sent, it is automatically delivered to the inbox of the recipient. If the user is offline, the collaboration server will queue the message until the next login. Collaboration messages are specific to the planning domain; they represent a transition in the workflow life cycle of a planning task, including actions such as "accept," "submit," or "approve." Each collaboration message contains an embedded plan, a portion of which is designated as the fragment of interest. Figure 32 illustrates this life cycle as a state transition diagram. Finally, system messages are used to communicate logins, logouts, and error messages, and are typically viewed only by an administrator.
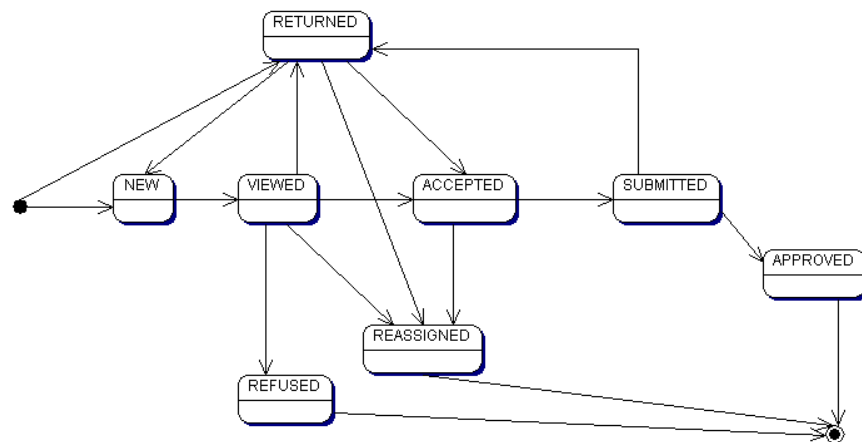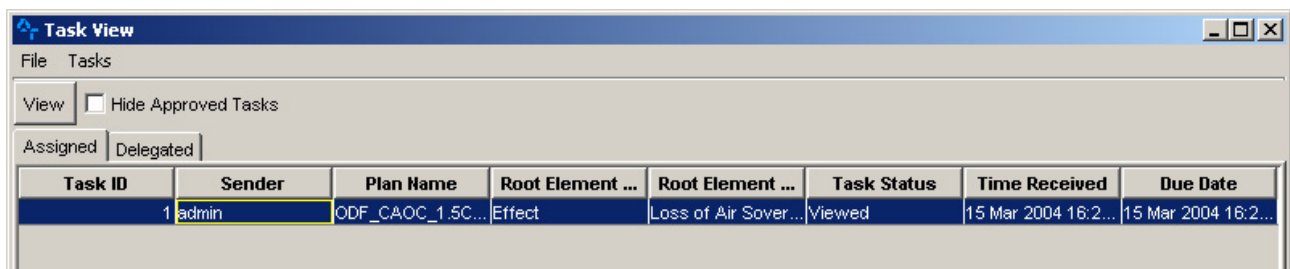
Figure 32. State transition diagram representing a planning task life cycle

Message reliability and planning task persistence are achieved through a combination of the bottom and middle tiers. The JBOSS server provides a remote message queuing service allowing messages to be sent across a local area network (LAN) and delivered to queues associated with registered users. If a message cannot be delivered, JBOSS will retry the delivery process and eventually notify the sender if the problem persists. The collaboration server provides persistence by efficiently storing user accounts and outstanding planning tasks in the form of serialized Java objects. If a client were to unexpectedly crash, the user's work will not be lost since it was backed up by the collaboration server.

The SDT provides four main tools for collaboration:

- *Account Manager* – Enables an administrator to add, modify, or delete user accounts

- *User List* – Enables a user to see who is logged into an SDT application on the network

- *Message View* – An email-style tool showing all received and sent messages, highlighting unread messages in bold. The user may double-click to view a message or preview it in the lower pane.

- *Task View* – A workflow-oriented tool with separate tabbed panes showing tasks delegated to other users and tasks assigned by others. The user may double-click to browse details of a single task.

Here, we will focus on the Task View to illustrate a typical life cycle of a planning task. A user may assign a portion of a plan to another user by selecting the top-most node to be delegated in the SDT's tree view and invoking the "Delegate" operation in the tree view's popup menu. This invokes a dialog in which the requester chooses an appropriate recipient and optionally types in guidance. Upon completion, this initiates an "assign" message to the recipient containing the entire plan, with the assigned node marked as the root. In this example, a plan fragment rooted at the effect "Loss of Air Sovereignty" has been assigned.

Figure 33. Task View showing a new delegated planning task.

At this point, the message appears in the recipient's Task View as a new delegation, shown in Figure 33. When the recipient reviews the guidance in the message and opens the embedded plan, it restricts read/write privileges to plan elements at or below the delegated plan node. The rest of the plan elements' tree view icons appear with red "X," indicating their read-only status, as shown in Figure 34. The recipient then chooses "accept" to notify the sender that the task was accepted and proceeds to elaborate the assigned plan fragment by adding new plan elements or editing attributes of existing plan elements. When the recipient is finished, he or she sends a "submit" message, which embeds the edited plan fragment. Finally, the original requester may view the submitted fragment in the context of the overall plan. At this point, the assigned root plan node and all its descendents are shown as read-only. If the requester chooses "accept," they change back to normal read/write status, and the workflow of the planning task ends. Alternatively, the requester may choose "reject" to send the task back to the recipient, providing an associated text message with corrective guidance. Many such planning tasks may be simultaneously delegated, and a recipient may choose to delegate lower-level fragments to yet another specialist.
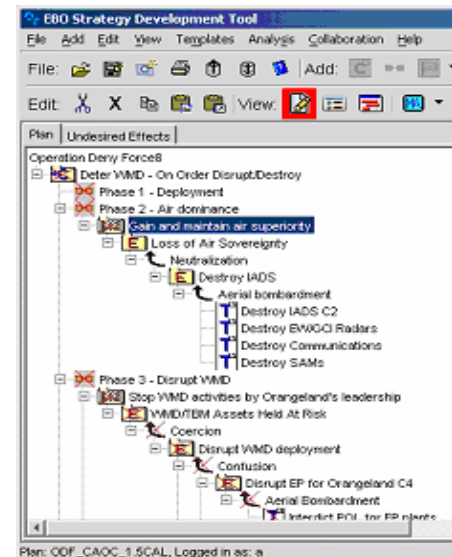
Figure 34. Delegated plan fragment, with external nodes marked read-only.

Real-time Collaboration

Our approach to real-time collaborative planning was based on a prior tool developed under DARPA's Active Templates program called Adaptive Planning System (APS), which included real-time collaborative capabilities for a different domain (Special Operations Forces) (White, 2004). During JEFX '04 Spiral 2 we demonstrated this tool to strategy team users, who verified that it embodied the above features. Thus, APS served as a model for incorporating similar capabilities in SDT. Due to the limited development time remaining (three months), users agreed that SDT could employ a coarser-grained level of collaboration than APS by batching updates at the level of an entire plan node rather than individual fields or attributes.
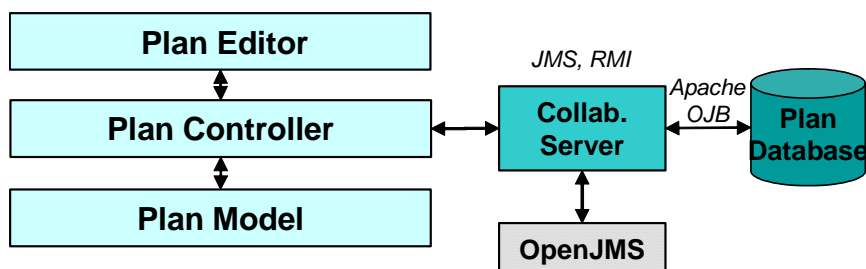
Figure 35. Real-time Collaboration uses a three-tier architecture based on J2EE standards.

52

The new architecture, shown in Figure 35, was based on Java2 Enterprise Edition (J2EE) standards and emerging technologies. We augmented the original file-based persistence with a database tier underneath the collaboration server to maintain a single common persistent storage of plans. The role of the collaboration server evolved from tracking delegated fragments toward logging, persisting and publishing incremental plan node changes. As these events are published, notifications are pushed to the application via OpenJMS subscriptions, so that SDT can update its views to display the changes. Because SDT's application layer had already been designed using a Model-View-Controller[1] pattern, these change events could be handled centrally in the Controller, allowing views to update themselves in the same way that they would with local updates.

To persist plans to a common database, we chose MySQL both for its lightweight open source heritage and the fact that it was already used in SDT as a target system database. An internal Persistence Manager component within the application server handles the translation of plan changes (adds, deletes, pastes and updates) into database operations as well as the conversion of database plan content into Java objects. Apache's OJB library, based on the Java Data Object standard, allowed us to declaratively map our existing Java plan objects and attributes to database tables and columns, providing object-to-database conversion methods for free. This significantly minimized the coding required for the Persistence Manager, allowing us to focus effort on application-level collaboration functions.

While user interface transparency was a first principle, we needed to enhance certain views with new features designed to prevent multiple users from trying to update the same plan node simultaneously and to keep them informed of changes as they occur. The following enhancements were made to the SDT Tree View and Properties Editor:

1. When a user begins typing a change in the Properties Editor, the node is immediately locked by that user.

2. Nodes locked by other users are marked with an "X" in the tree view, and the username of the locking user is shown in parentheses after the node name (see Figure 36).

3. Users may browse the properties of locked nodes in the Properties Editor in "read only" mode, with editing of field values disabled.

4. When another user checks in a change to a plan node, the Tree View shows the plan node's name in boldface font until the current user clicks on the node to inspect the change. This feature was based on modern email systems that mark new unread messages in bold.
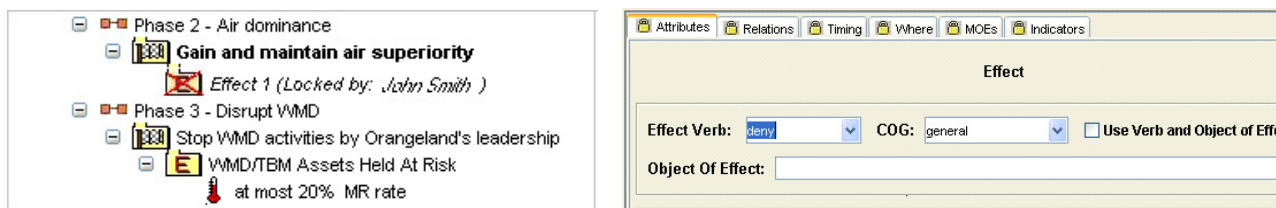
Figure 36. In real-time collaboration, users lock and edit individual plan nodes simultaneously, with synchronized plan updates.

---

[1] See Sun Microsystems, "Java Blueprints," http://java.sun.com/blueprints/patterns/MVC-detailed.html

53

In addition, to enable collaborative Mission Analysis, the tabbed pane editors featured in the Mission Analysis and Commander's Intent dialogs were modified to allow tab-by-tab locking. Here, as soon as one user adds a new statement or begins typing in an existing statement, the tabbed pane containing the statement is locked. A lock icon appears on that tab and other users can read but not write within that tab. When the change is checked in, the content of the tabbed pane for the other users is automatically updated with the new information.

Our redesign intentionally retained the SDT's original stand-alone plan editing and file-based persistence capability to allow for situations where collaboration is not needed or an outage occurs on the database machine. This required the ability to switch between file-based offline planning and collaborative planning. To achieve this, we added new File menu operations for uploading the current plan loaded in SDT to the database and downloading a database plan. After either action, SDT switches to collaborative planning mode. Another menu item, "Work Offline," was added to switch to file-based mode in case the collaboration server or database goes down. Due to the limited time, we chose not to implement a capability to merge changes made while working offline back into the database version of the plan.

3.7.3 COA Assessment

Some of the final native SDT features added for JEFX '04 were aimed at the Operational Assessment team during Main Ex (the final spiral held in July 2004), since they would finally be assessing the execution of the plans developed during previous spirals. These assessment features include a new COA Comparison Matrix, new Plan Editor tabs for planned vs. actual assessment milestones, and progress indicators in the Timeline.

COA Comparison Matrix

The COA Comparison Matrix, based on the COA Comparison step of the JAEP process, is a new dialog available in SDT's Analysis menu. It shows a formatted table in which users may enter rankings for a number of Criteria (such as acceptability, suitability, completeness) per each phase per each COA. The columns show each COA, and the rows are shown as two tree levels, the outer showing phases, the inner showing the criteria. A set of rows labeled "Totals" is provided at the bottom so users may see immediately which COA ranked highest overall. The COA Comparison Matrix allows users to tailor the criteria and Phase labels via a separate dialog. When a user saves and loads the main plan, SDT also saves and loads the state of the COA Comparison Matrix to the XML plan file. It does not persist to the database during collaboration mode. Figure 37 shows a screenshot of the COA Comparison Matrix taken from the SDT User's Manual.

Figure 37. The COA Comparison Matrix aids users in scoring each phase according to a standard list of criteria to choose the best overall COA.

COA Assessment Editors

The first major spiral in which players operated the initiative tools at JEFX was Spiral 2, in March 2004. Activities in the Strategy Division were limited to COA planning only, since there was no pre-existing plan for execution yet. However, in Spiral 3 and Main Ex the Operational Assessment team would be assessing the execution of the previous spiral's plan developed by the Strategy Plans team. Thus, AFRL approved work on a baseline assessment capability in SDT aimed at helping operators track progress through plan execution (as opposed to the more sophisticated assessment of COA or effect success likelihood performed in OAT). To this end, BAE Systems added new COA Assessment Editor tabs to the Properties Editor for plan nodes at all levels to provide a means of capturing and viewing plan progress.

In the final implementation, two new assessment tabs were added. The first tab, Planned Assessment, is used during planning and allows users to specify prior desired milestones for

each plan node in terms of Percent Complete at a specified date in the operation (in terms of C or D Day or calendar date). The Actual Assessment tab (shown in Figure 38) is used during execution to record actual progress in terms of Percent Complete on a particular date. It also contains a Schedule Status field for estimating whether the plan node is Ahead, Behind, or On Schedule. A Remarks column is included in both tabs for entering textual notes. As with other Properties Editor views, any change turns on the "dirty flag" on the Apply button, and changes are saved to memory when the user clicks Apply and saved to the plan file when the user saves the plan (in collaborative mode they are saved to the database when the user clicks Apply).

Figure 38. Assessment tabs allow users to track progress through the plan.

The original design for the Actual Assessment tab was to use a rollup algorithm to automatically compute % Complete and Schedule Status for all plan node levels above Tasks, which would be manually entered. Unfortunately, we did not have sufficient time to implement this algorithm, so all levels required manual entry for the final SDT 2.0 Main Ex release.

Timeline Progress Indicators

In coordination with the COA Assessment tabs, we developed a major enhancement to the Timeline Editor in which the % Complete level of each plan element would be displayed as progress indicators within each Gantt bar. The design for this was based on a similar technique used in Microsoft Project. After applying an entry to an assessment tab, switching to the Timeline Editor would show a progress indicator for that plan element starting from the left side of the Gantt bar, with length proportional to the latest % Complete entry for that element. In addition, we added a vertical yellow line showing current time. This made it easy to determine which tasks were behind schedule by looking for progress indicators that ended to the left of the vertical line. Figure 39 shows a screenshot of an early mockup of this display, which very closely matches the final implementation.
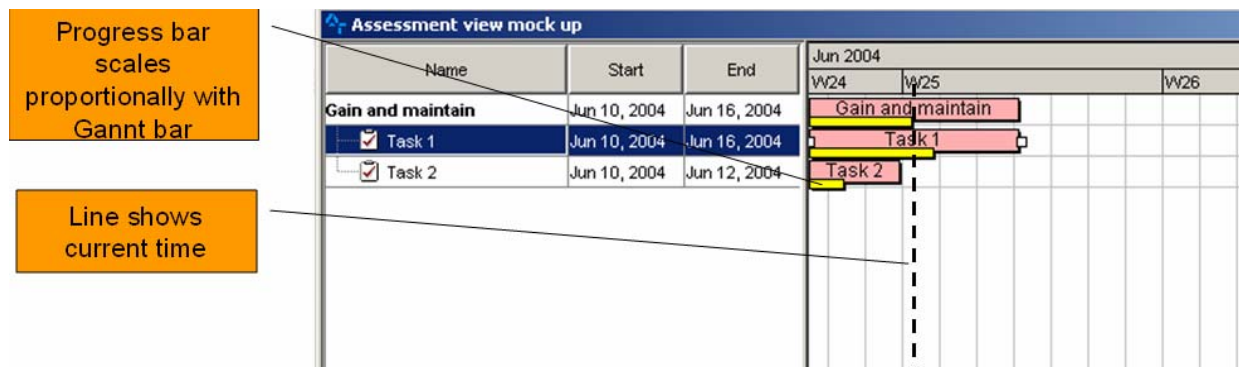
Figure 39. Timeline Editor was extended to display progress indicators for assessment.

Other Enhancements

Several other more minor enhancements were made based on feedback from the 8th AF players during early spirals. The full record of these changes have been maintained in our project Software Task Order (STO) database. They include but are not limited to the following:

- A new "Undesired Effect" plan element representing potential negative effects arising from tasks or intermediate effects. The new node type may be used wherever normal effect nodes are used, and is available within the Add toolbar and Structural Editor.

- A new Strategy-to-Task planning mode, available via the Edit menu, which turns off the normal parent-child restrictions used with the EBO core ontology. During this mode, users may add any node type under any other, allowing them to build plans using older methodologies like strategy-to-task.

- In COG Articulator we added the ability to import existing model fragments on top of the current model. This works somewhat similarly to Mission Templates in the Plan Editor, except that it is left to the user to connect the new fragment to a node in the existing model by drawing COG links.

In addition to these enhancements, numerous bug fixes were performed in between spirals, which have also been logged within our STO database.

3.7.4   External Interfaces

Throughout both the original ATD spirals and JEFX '04 preparation, BAE Systems developed several salient external interfaces to other Air Force and EBO-related systems and worked with several co-contractors interested in integrating with SDT. We have already described the interfaces to CAT and Endstate. The following subsections describe additional interfaces to EBOWS, DCOAD, and TBMCS.

EBOWS

The EBO Wargaming System (EBOWS) was adapted for the EBO ATD by L-3 Communications, and was based on an earlier force-on-force military simulator called THUNDER. EBOWS was part of the EBO ATD and the original AFRL PBA/EBO JEFX '04 initiative architecture. Throughout late 2003 we held numerous teleconferences with EBOWS

technical representatives to answer their questions about the SDT plan representation. We also made slight modifications to the plan ontology and Operation Deny Force sample plan to help them exercise their wargaming vignette. In 2004, they had demonstrated the ability to read in SDT plans, mainly at the tactical task and target level, and adjust pre-scripted wargaming scenarios based on the targets and task start dates in SDT. No further integration work was done for JEFX '04, because EBOWS was not included in the final initiative architecture.

DCOAD

Another tool BAE Systems had been developing under AFRL 6-2 funding that played a secondary role in JEFX '04 was the Dynamic Course of Action Decision (DCOAD) tool. The main purpose of this tool was to help ISR and targeting analysts determine gaps in ISR or strike coverage due to Time Sensitive Targets (TSTs). However, during JEFX '04, the primary capability of interest was DCOAD's map-based and tabular displays of IPB data. DCOAD had been built to import XML data files from AFRL's A2IPB tool, which was on the PBA side of the AFRL initiative. By incorporating the DCOAD tool within SDT via a new launch button, users in the Strategy Division would have direct access to the IPB data produced by the ISR Division, to use as a valuable reference during strategy planning. The DCOAD map layer was integrated within the existing Target Set Tool map display in SDT. DCOAD's tabular display of IPB data was included in the main DCOAD window launched via the DCOAD button on the SDT Analysis toolbar. During Spiral 3, we worked with ISR liaison operators in the Strategy Division on a manual process for subscribing to A2IPB change files via their web-based interface, and importing the files into DCOAD. Figure 40 shows an example of the DCOAD map display with IPB overlay.
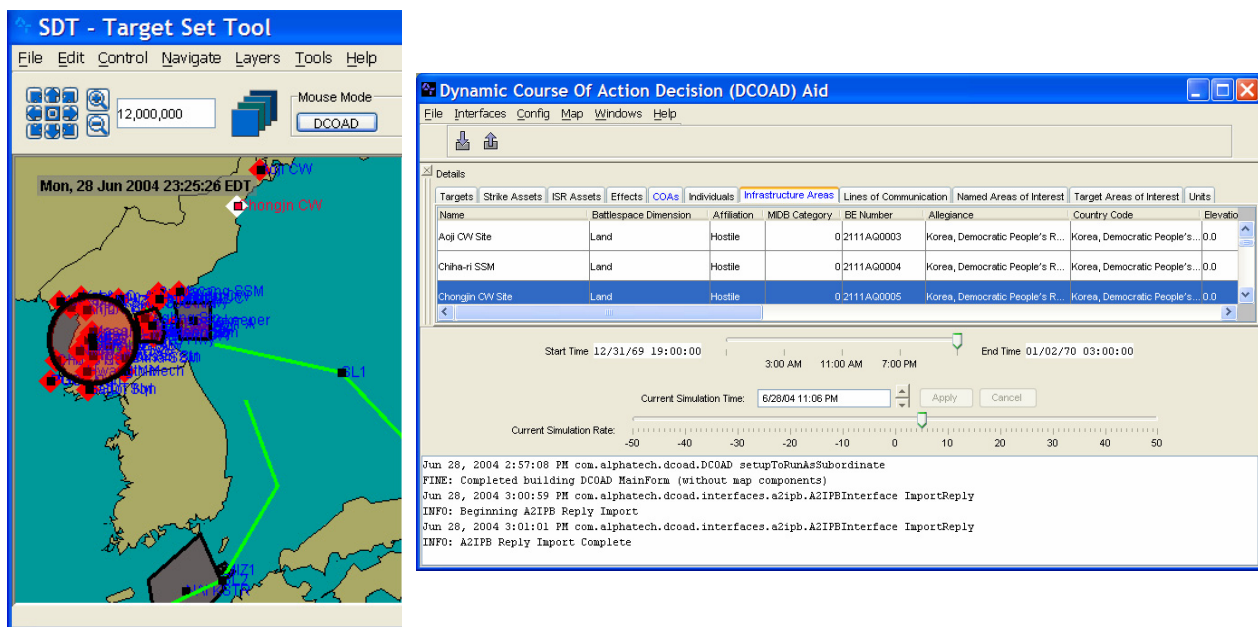
Figure 40. DCOAD map display and table view of IPB data.

TBMCS Strategy Management Service

The most important external integration goal for JEFX '04 was the interface with the TBMCS 1.1.3 Strategy Management Service (SMS) developed by Accenture. BAE Systems' integration efforts were funded by Electronic Systems Center (ESC), also the sponsor of SMS. SMS was intended to be a centralized repository of plans developed by the Strategy Division, using a common plan schema that had incorporated aspects of the multiple strategy tools being used at JEFX '04. Each strategy tool developer, including SDT, was required to develop translators from their native plan format to the SMS format specified in their Web Service Description Language. While there was considerable jockeying for visibility among these tools prior to JEFX, the final concept of operations shown in Figure 41 centered on SDT (the only strategy tool playing in an initiative) as the primary general strategy tool. General Dynamics' Information Warfare Planning Capability (IWPC) would be used for Information Operations planning in parallel. Prior to handoff to targeting, Accenture's Strategy Planning Tool (SPT) would be used to manually merge IWPC plan fragments with the mainline SDT plan. Finally, the Joint Targeting Toolbox (JTT), which included a strategy-to-task editor, would read in the merged plan in order to append target lists and associate targets with tasks and objectives (SDT's similar targeting capability was not used at JEFX '04 because of this pre-arranged process).
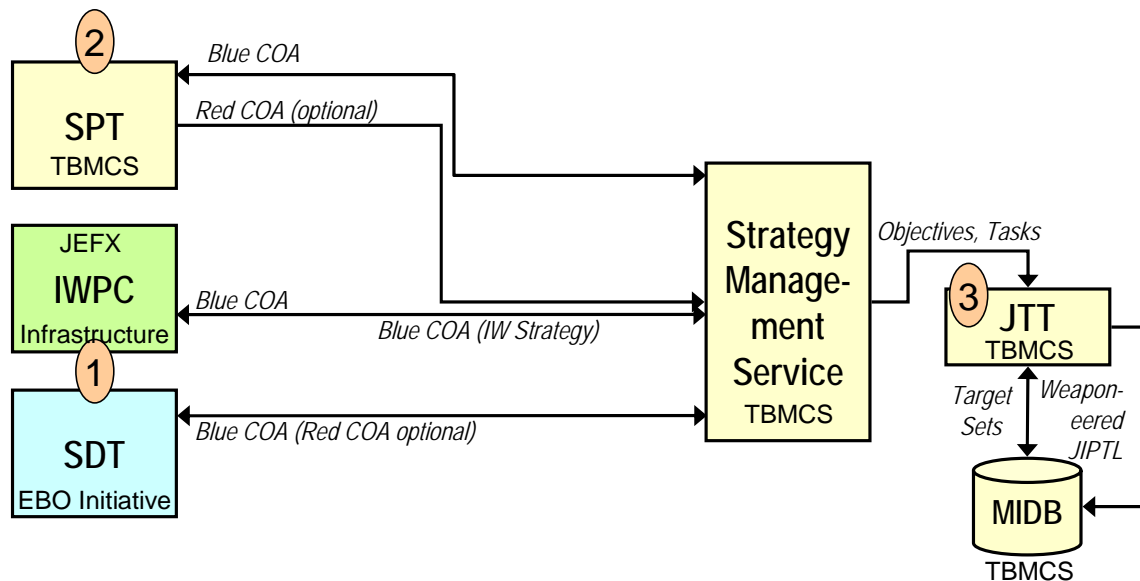
Figure 41. Strategy tool architecture and dataflow for JEFX '04.

Our approach to the SMS interface involved two tasks, reflected in the integration architecture diagram in Figure 42:

1. Develop a translators using XSLT, a concise language for transforming XML documents, for forward and reverse conversion between the SDT and SMS plan formats.

2. Develop a SOAP-based client to access the SMS web service.

Task one was begun well in advance of JEFX '04 Spiral 2, allowing us time to coordinate with Accenture on questions about the SMS plan schema. The primary challenge was transforming from SDT's hierarchical plan schema to SMS's relational schema, which called for

separate flat lists of plan elements and parent-child relations. This required subtle XSLT techniques for keeping state of previously read XML elements from the flat list in order to create the hierarchy later. In the end a series of incremental XSLT transform files were used, with the reverse and forward transforms kept in separate directories. Accenture provided runtimes of their SPT application to allow us to test locally whether the translated SMS plans were valid, although this was complicated by slight differences between SPT and SMS formats.

Task two was significantly delayed because a viable client API with working sample code was not released by Accenture until early February 2004, a month before Spiral 2. Despite the short timeline, we were able to implement a working SOAP client and successfully translate and upload SDT plans to SMS by Spiral 2. The reverse translation still had significant bugs, however, which were resolved later in time for Spiral 3. Another issue that arose during Spiral 2 and 3 was the slow translation time, taking up to twenty minutes depending on network congestion. Prior to Main Ex we worked with a senior XML specialist at BAE Systems to optimize the XSLT code to achieve an order of magnitude improvement in translation time, which was well-received during Main Ex.
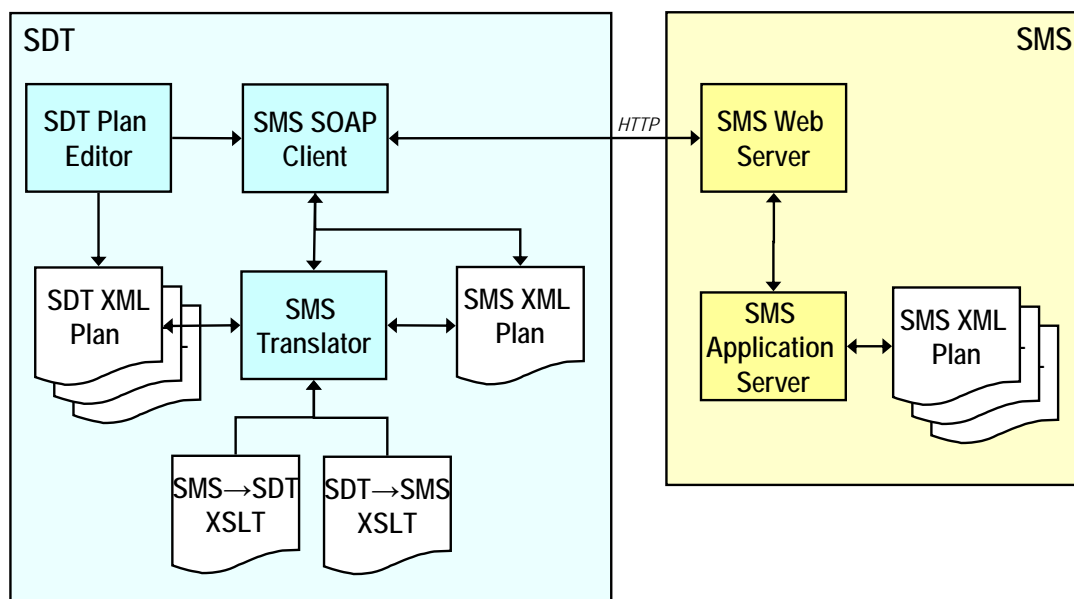
Figure 42. Integration architecture for the SMS interface.

More detail on the SMS interface can be found in the SDT Interface Control Document in Appendix C.

Other Interfaces

During early JEFX '04 preparations, AFRL and BAE Systems held system engineering discussions regarding integration with several other potential AOC tools, including:

- JTT – The ability to automatically export target sets chosen in SDT via Endstate's query tool or option generation would eliminate manual duplication of entries and ensure effects-based strategy and target-to-task linkages from the strategy team would be preserved for the targeting team.

60

- Joint Defensive Planner (JDP) – This tool would be used to elaborate the blue defensive strategy.

- IWPC – This tool would be used to elaborate strategy relating to Information Operations.

JDP and IWPC integration efforts were eventually abandoned because JDP was not included in the final initiative architecture, and an indirect interface with IWPC was achieved via SMS. We did hold discussions with AOC SMEs about options for sending targets to JTT. A prototype XSLT translator had been developed for exporting SDT targets and target-to-task linkages to JTT's XML target list format, but because we did not have the latest version of JTT in-house we were not able to sufficiently test it prior to JEFX '04. The issue was largely moot because SDT's targeting capability and Endstate's analysis tools were not used at JEFX other than for stand-alone evaluation.

3.8   FINAL SDT ARCHITECTURE

Figure 43 shows the final software architecture for SDT 2.1. The architecture shows two levels of decomposition. The outer boxes represent *components* (also called Computer Software Components, or CSCs), while the inner boxes represent *units* (also called Computer Software Units or CSUs) within the components. Some components do not contain internal CSUs. Boxes shown in gray represent external tools, services, or databases external with which SDT interfaces.
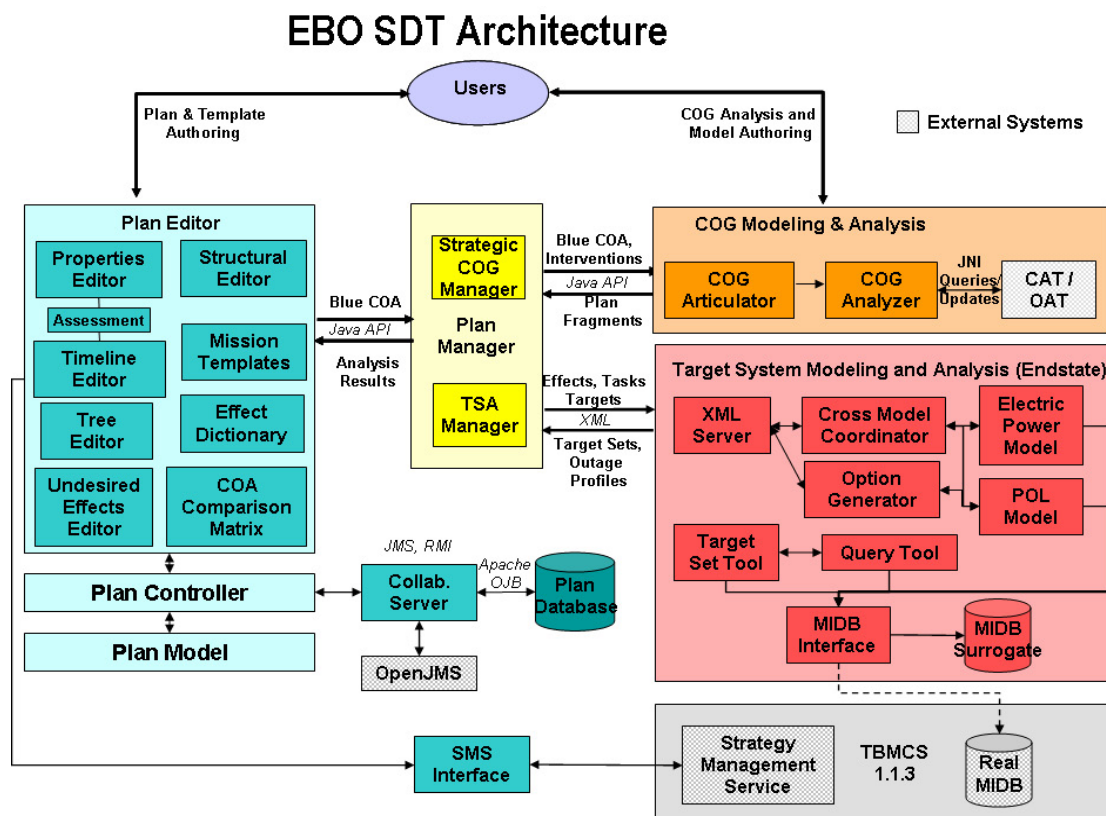
Figure 43. Final SDT architecture.

Detailed descriptions of the architectural components are given in the SDT Software Design Document (Pioch, 2004). Inter-tool communication technologies include direct Java API, XML messaging, Java Messaging Service (JMS), Remote Method Invocation (RMI) and Java Native Interface (JNI).

SECTION 4

RESULTS AND DISCUSSION

Throughout its period of performance, SDT participated in numerous integration activities, end-user experiments, and demonstrations. This section describes the most salient of these events, including discussion of primary outputs, findings, and lessons learned.

4.1   CROSS THRUST DEMONSTRATIONS

Throughout the first two full years of the AFRL EBO ATD program, SDT participated in a multi-disciplinary "Cross Thrust" demonstration showcasing past and present tools developed under the EBO jumpstart and ATD as well as tools from other related programs. The primary audience of the Cross Thrust demonstration was the Air Force Scientific Advisory Board (AF SAB), who received a demonstration during a visit to AFRL in November, 2001 and during a special conference at Langley AFB in March, 2002. Further extensions to the Cross Thrust demo culminated in a demonstration at the final EBO ATD PMR in January, 2003.

4.1.1   AF SAB Demonstrations

During Year 1, SDT and Endstate were demonstrated both to the AF SAB and to the audience of the November 2001 EBO PMR as stand-alone prototypes, since we had not had sufficient time to tightly integrate them with each other or with other EBO jumpstart tools. The SDT demonstration consisted of a stand-alone natural language plan editor with a sample air campaign plan based on Operation Deny Force. We also demonstrated the strategic COG models we had built in CAT, showing the probability of POL supply with vs. without interventions against key POL nodes (see Figure 27). The Endstate team demonstrated their preliminary TSA algorithm, showing a table-based display of outage profiles predicted for the EP network reflecting the enemy's workarounds to the blue strike on the EP network. While all three capabilities were running stand-alone, the SDT plan, CAT model, and Endstate TSA all adhered to a common storyboard we had developed based on the EBO challenge problem Operation Deny Force. In this storyboard, the top-level desired effect in the SDT was to "disrupt TBM development and deployment." Based on the POL value-chain COG model in CAT, the planner decides on a strategic-level task "degrade LOCs" in order to achieve the intermediate effect of "disrupt POL supply to TBM units." Using Endstate's LOCS Workarounds Reasoner (not included in the final system), the planner over several trials determines which LOCS targets will lead to a sufficient delay in POL supply logistics, based on the simulated enemy repair and logistics workarounds. Finally, the planner approves the target set and the system appends an operational-level augmentation to both the SDT plan and the CAT COG model reflecting the chosen targets (two bridges) and the intermediate operational-level effects leading up to degradation of LOCs.

In the months following the first SAB demonstration, we participated in an internal AFRL integration activity named "Cross Thrust," reflecting an operational architecture cutting across multiple programs and research areas. The Cross Thrust would extend the EBO jumpstart architecture, which featured CAT, an unclassified JTT emulator built by AFRL for weaponeering, and a scheduling tool built by Steve Smith's group at Carnegie Mellon University

(CMU). The concept of operations for the Cross Thrust inserted SDT at the front end of the process:

1. Demonstrate pre-authored ODF_CAOC.xml plan in SDT.

2. Export plan from SDT to CAT to perform prediction of timing and probability of COA success without targets.

3. Import target nodes into CAT underneath existing task nodes via AFRL's JTT emulator. Each target includes several choices for available weapon resources, each with different P(kill) based on the Joint Munitions Effectiveness Manual (JMEM) historical estimates.

4. Using worst-case estimates of P(kill) for targets, reanalyze the plan in CAT for a new estimate of P(COA success).

5. Export the plan from CAT to the CMU Scheduler for detailed scheduling and weapon selection. Examine the resulting schedule in the Scheduler GUI.

6. Reanalyze the plan in CAT a third time, based on the revised P(kill) for the selected weapons.

By March 2002, BAE Systems had completed the SDT-CAT interface to support the ability to export blue COAs to CAT for causal analysis, and had worked with AFRL and other Cross Thrust participants to achieve the above end-to-end demonstration thread. We also updated our scenario to elevate the top-level objective in the plan to be more consistent with the main goal of Operation Deny Force, "Deter WMD – on order Disrupt/Destroy WMD." We developed a two-phase plan in which Phase 2 Air Dominance was elaborated with an air_superiority mission template developed by SME Buster McCrabb, and Phase 2 Destroy WMD contained direct strikes against various classes of WMD targets. This plan, along with the timing (strike times, delays, and persistences) and causal strength estimates, was exported to CAT via SDT's Analysis toolbar. CAT then predicted the probabilities of success over time of the overall COA and its underlying objectives and effects. These results were displayed in the probability profiles view and the assessment view. By this time, SDT had also established an XML-based messaging interface to the Endstate subsystem, but the Endstate user interface was not sufficiently mature to include in the SAB demo.

That same month, we presented this more refined EBO demonstration during an AF SAB conference at Langley AFB in Hampton, VA. Although we had loaded the entire suite of Cross Thrust software onto our demo laptop, time constraints led to the decision to limit the scope of the demo to the first two steps, review of the plan in SDT and export to CAT for causal analysis. The demonstrations were attended by SAB members, other technologists, and AFC2ISRC representatives responsible for future AOC requirements. Several of the attendees recognized the need for the COG/TSA part of the SDT, which was not included in the demo, but was covered on request in slide material. One SAB member who had criticized the fall presentations at AFRL praised the progress reflected in the presentation. Another commented that the Air Force has always done effects-based targeting, just not by that name. Yet another commented that using ontologies to represent new concepts like EBO can lead to configuration management issues, as the ontologies tend to be brittle as the concepts evolve.

4.1.2   EBO Program Management Reviews

In preparation for the April 2002 EBO PMR we developed a more extensive Leadership COG model in CAT in which the top-level goal of the enemy is to Deploy WMD, and the model drills down to reveal the dependencies necessary for the leadership to be willing to deploy WMD and to be able to deploy it (see Figure 26). We then implemented a new bi-directional interface to CAT in which all red COG nodes in the CAT model could be imported to SDT, selected as objects of effects or tasks via the "Affected COG Node" statement editor, and then the blue tasks from SDT could be exported to CAT with inhibitor linkages connecting them to the directly affected red COG nodes (see Section 3.6.1). We had also achieved a preliminary XML-based messaging interface with the Endstate subsystem in which SDT sends a plan summary message and target list message to Endstate to initiate the workarounds TSA simulation. However, at this point the target list was hardcoded, and the Endstate results were only shown in a rudimentary map display in Matlab®, not sent back to SDT. These capabilities were demonstrated jointly by the SDT and Endstate teams during our briefings to the April 2002 PMR audience. In addition, we demonstrated the Cross Thrust tools described in the previous section during a separate poster session. In addition to insightful questions and suggestions, we received generally positive feedback on the demonstrated concepts and prototypes. In particular, a visiting JWAC representative expressed interest in the idea of building causal COG models to explore the impact of blue interventions.

By the final EBO PMR in January 2003, after the completion of the Target Set Tool and Query Tool, we had achieved for the first time a complete end-to-end integration of all three systems, SDT, CAT, and Endstate. We had also successfully incorporated the ISX Effector editors as alternative views of the EBO plan at varying levels. We refined the ODF_CAOC demonstration plan to include five sequel phases and one branch phase:

- Phase 1 – Deployment – not used

- Phase 2 – Air Dominance – import air_superiority mission template

- Phase 3 – Disrupt WMD – reflected indirect targeting of EP nodes to disrupt WMD facilities

- Phase 3 Branch – Destroy WMD – on indication of WMD deployment, destroy WMD facilities

- Phase 4 – Compliance – not used

- Phase 5 – Redeployment – not used

Phase 2 was used to demonstrate the ease of incorporating mission templates as pre-defined plan fragments. Phase 3 was used for two purposes. The strategic-level effects and tasks were based on an intervention analysis performed in CAT, in which three alternative interventions, Destroy LOCS, Destroy Raw Materials, and Destroy EP, were compared in terms of their impact on the enemy's WMD production. After the Destroy EP was shown to be most effective, another more detailed version of Phase 3 elaborated the plan down to operational-level EP targets and effects and Endstate was used to confirm that these targets indeed produced the desired outages on the WMD production facilities. Finally, Phase 3 Branch illustrated the power of the EBO ontology to represent contingency planning, similar to the procedure for COA wargaming, in which we plan for the event that the enemy deploys its WMD out of perceived pressure that it

will lose them. This phase, taken from a portion of the original Cross Thrust plan, directly targets multiple classes of WMD facilities with the effect of destroying their WMD capability indefinitely.

The SDT-Endstate portion of the demonstration had become significantly more cohesive due to the Target Set Tool and Query Tool. The Target Set Tool's map display was a much more intuitive visualization of the EP and POL networks than the original Matlab® plot. The Query Tool, combined with the Direct and Affected Target list panes, provided a way to dynamically select candidate target sets and incorporate them into the plan, eliminating the earlier hardcoded target list. Finally, we had completed the bidirectional XML interface with Endstate, such that the results of the Endstate TSA were received by SDT and available for inspection in outage profile graphs on highlighted target nodes in the Tree View.

The SDT and Endstate teams again jointly delivered a briefing and demonstration of the above functionality at the January 2003 EBO PMR in Hampton, VA. BAE Systems and AFRL also delivered a reprise demonstration of the Cross Thrust tools described in the previous section. Feedback was again primarily positive from the attending representatives of the end-user community and other government researchers. At the end of the PMR, BAE Systems participated in a sidebar discussion with ISX and Accenture regarding requirements for a future operational AOC strategy tool sponsored by AFC2ISRC and managed by ESC/Hanscom AFB. We also attended a brainstorming meeting to plan for a small-scale EBO process workout in April 2003, which would prove to be an important stepping stone to JEFX '04.

4.2   EBO WORKOUT

In early 2003, BAE Systems participated in planning teleconferences and meetings to help organize a special knowledge acquisition activity called the "EBO Workout." The goal of the activity was to observe an assembled team of active and retired senior AOC operators walking through a modified effects-based variant of the Joint Air Estimate Planning process (EB-JAEP), using a fictional scenario based on Millennium Challenge 2002. Dr. McCrabb presented background briefings on EBO theory and prepared an outline for the EB-JAEP. David Hess of AFC2ISRC moderated organizational meetings, and led the formation of the various teams for players, assessors, controllers, and technology developers. This latter team consisted of the BAE Systems leads for SDT, representatives from ISX, Accenture, and General Dynamics, and government sponsors from AFRL and ESC. It was decided early on that because the main goal was to identify improvements to existing planning doctrine and process, the EBO tools would not be used by the players. Rather, the technology teams would "shadow" the players as they walked through the process, building up plans and models in the tools corresponding to the paper products developed by the players.

Throughout the EBO Workout, we gained valuable insight into the challenges and creative solutions proposed by the players as they worked to put the EB-JAEP into practice. During the first two days, they focused on Mission Analysis, which included IPB and COG analysis. This was an excellent opportunity to shadow the results of their COG analysis in CAT. In fact, the SDT POC worked directly with player Duck McSwain to use CAT to author a leadership model for the enemy state Califon. This model was not a true causal model because the nodes were not stated as Boolean true/false conditions, hence it would not make sense to quantitatively analyze it in CAT. Nevertheless, it effectively conveyed using graphical nodes, arrows, and color schemes the relationships among the key leadership figures and their supporting actors (see

Figure 44). In this model, red-shaded nodes represent enemy actors, green-shaded nodes represent actors sympathetic to U.S. causes, and yellow-shaded nodes represent neutral actors. Gray nodes represent current roles or groups and white nodes represent possible desired post-conflict roles.
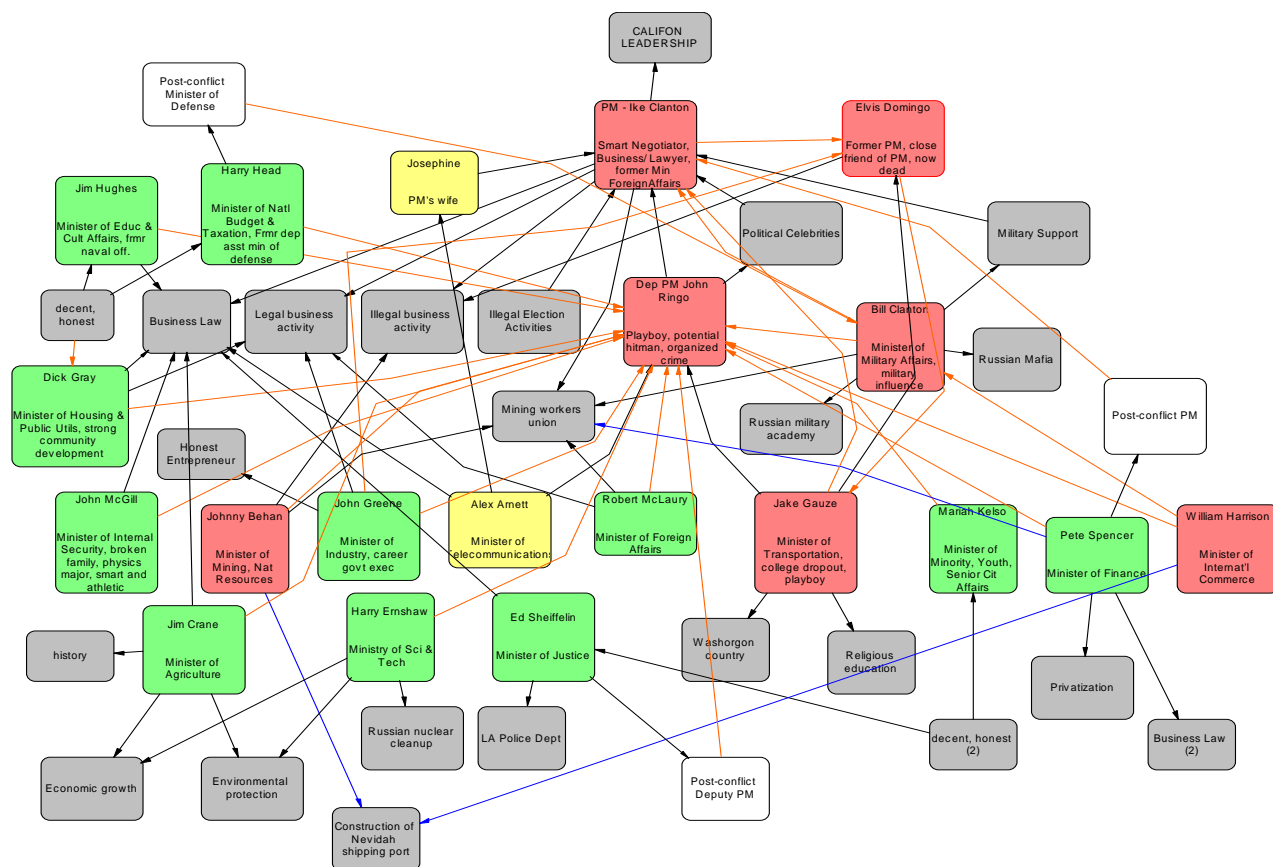
Figure 44. Califon leadership COG model authored in CAT at the EBO Workout.

During COG analysis, the players divided into separate teams to analyze different enemy COGs in parallel, then rejoined to brief each other on their work. At the request of senior AFC2ISRC Director Ken Calicutt (who was playing Deputy JFACC), BAE Systems used CAT to author a "shadow cross-COG model" based on the verbal and visual outputs of the players' analysis. This model synthesized causal fragments relating to leadership, fielded forces, economy, and communications, causally relating them to three top-level red objectives and capabilities. Unlike the leadership model, this model was intentionally built to preserve CAT's causal semantics, using true/false statements for each node and causal relationships consistent with the findings of the players. Figure 44 shows this model, along with the stable-state probabilities of each of these three red nodes. This represents a baseline probability of the enemy achieving their goals with no blue interventions.

During the last two days, the players turned their attention to COA development and analysis. Because this was their first exposure to EBO concepts they struggled initially with the semantics of effects and mechanisms and debated various ways to state effects in their COA. Eventually they settled on a statement template of the following format:

If we <Task>, then <Effect> will happen because <Mechanism>, as indicated by <MOE>.
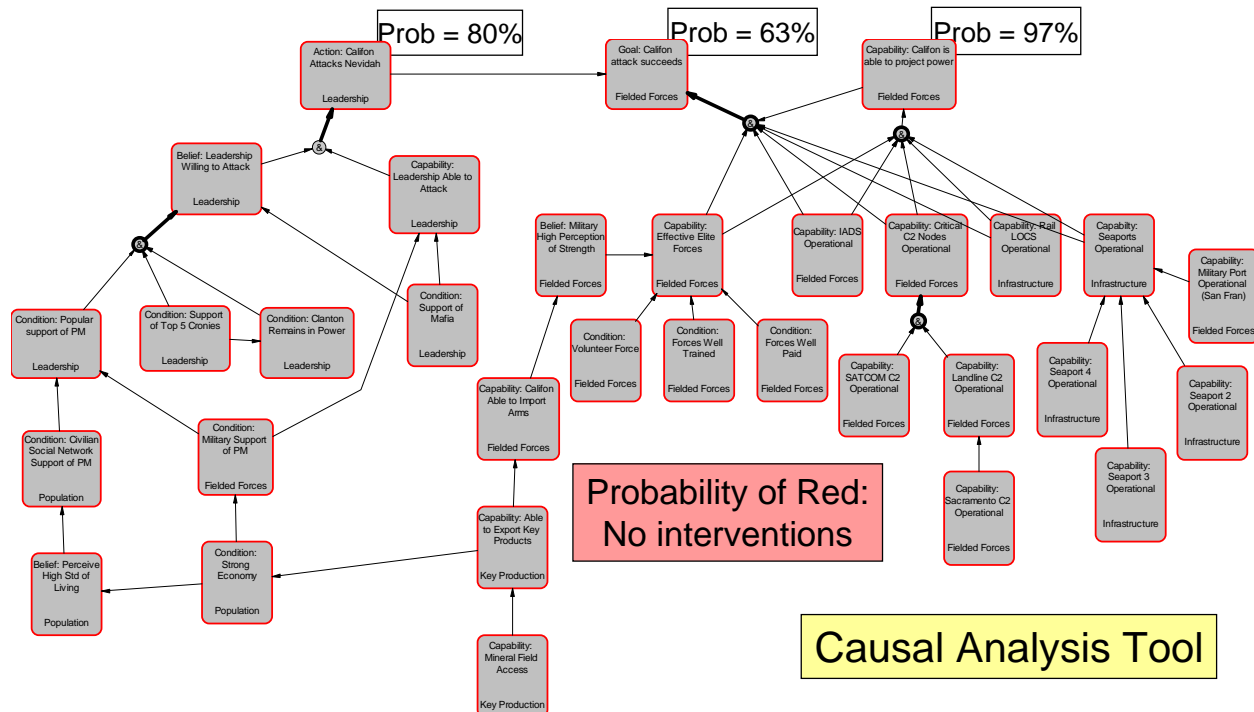
Figure 45. Califon cross-COG model with causal semantics for EBO Workout.

This format was straightforward to translate into corresponding SDT plan nodes. By the end of COA development, we had produced a detailed plan combining the effect-mechanism-task constructs developed by one of the subteams, shown in Figure 46.

68
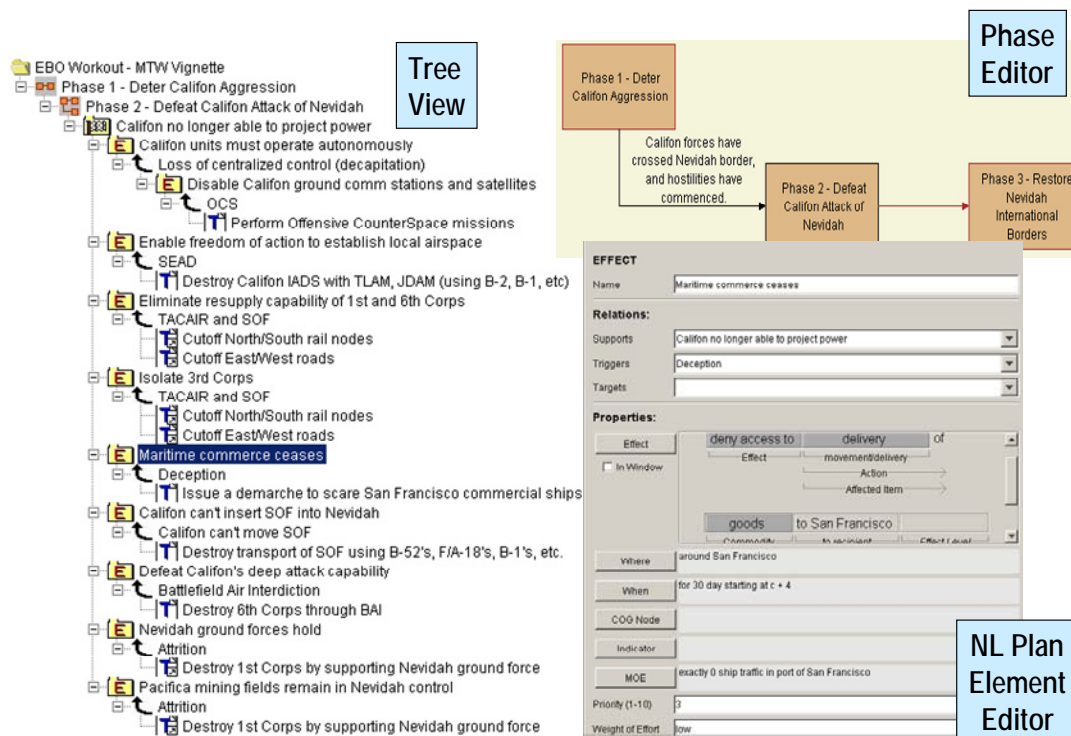
Figure 46. SDT "shadow" plan developed during the EBO workout.

Since the EB-JAEP called for comparing multiple COAs before selecting one for execution, the players worked in parallel teams on alternative COAs for the primary "Major Theatre of War" (MTW) scenario. As they worked separately, BAE Systems and AFRL pooled notes on the effects and actions arising from each COA. BAE Systems then extended the Califon cross-COG model in CAT to overlay a few representative tasks from the two main player COAs. We then reanalyzed the model under different combinations of interventions, with the resulting probabilities of the three top-level red goals shown in Table 6. COA1, which intervenes against the leadership, greatly decreases the probability of Califon attacking Nevidah. However, it does not impact Califon's ability to project power because the military can continue to operate independently. COA2, which intervenes against military C2 nodes, slightly decreases the military's ability to project power and succeed in an attack should it occur, but does not affect the probability that Califon will attack Nevidah. Due to AND groups built into the COG model, combining all interventions leads to significant reduction in all three red goals.

Table 6. Impact of alternative COAs on probability of red goals.

COA	Blue tasks	P(C. attacks N.)	P(C. attack succeeds)	P(C. able to project power)
Null	None	0.80	0.63	0.97
COA1	Decapitate Leadership	0.27	0.21	0.96
COA2	Destroy Satellite C2 Destroy Sacramento C2	0.79	0.57	0.78
COA3	All of the above	0.27	0.16	0.4

At the end of the EBO Workout, each technology team had the opportunity to outbrief their shadow activities and lessons learned to the entire group. In addition to walking through the above models and plans, BAE Systems prepared suggested tool requirements for potential implementation in SDT based on the observed planning activities. The most salient requirements are summarized in Table 7, along with the version in which some of the suggested features were eventually implemented.

Table 7. Suggested requirements for SDT and CAT based on the EBO Workout.

Tool	Activity	Requirement	Implemented?
SDT	Mission Analysis	Richer representation of Commander's Intent / Msn Analysis	SDT 1.4
CAT	COG Analysis	Import of causal model other sources, e.g. MS Excel.	-
CAT	COG Analysis	Causal linkage weights other than probabilities	-
SDT	COA Development	Taxonomy of mechanisms	-
SDT	COA Development	Capture undesired/negative effects	SDT 1.6

SDT	COA Development	Capture indicators of failure as well as success	SDT 1.4
SDT	COA Development	Support bottom-up and middle-out planning	SDT 1.6

4.3 JEFX '04

Soon after the EBO Workout, BAE Systems began working with AFRL to prepare a Phase I proposal for a PBA/EBO initiative in JEFX '04. This ambitious initiative called for integrating many of the EBO ATD tools with various fusion and situational awareness tools from AFRL-IFE, as well as a number of strategy and targeting tools included in the JEFX infrastructure. After the Phase I proposal was accepted, we helped prepare slides for the Phase II oral presentation at Hanscom AFB, which was delivered by EBO ATD Program Manager Dan Fayette. By summer of 2003 word came back that the initiative was accepted. Over the next few months, we worked with AFRL, Dr. McCrabb, and C4ISR architecture expert Lee Wagenhals to develop standard DoDAF OV- and SV- architecture diagrams, which would be required by the JEFX technology assessors. Figure 47 shows the near-final version of the submitted SV-1 functional architecture. Green borders and arrows denote tools and dataflow not actually used in the deployed JEFX initiative. However, some of the green-colored dataflows were achieved via human-in-the-loop manual entry, such as indicator collection tasking within MAS. Also note that OAT was used in place of CAT, but was still referred to as CAT in order to be consistent with early interactions with JEFX assessors.
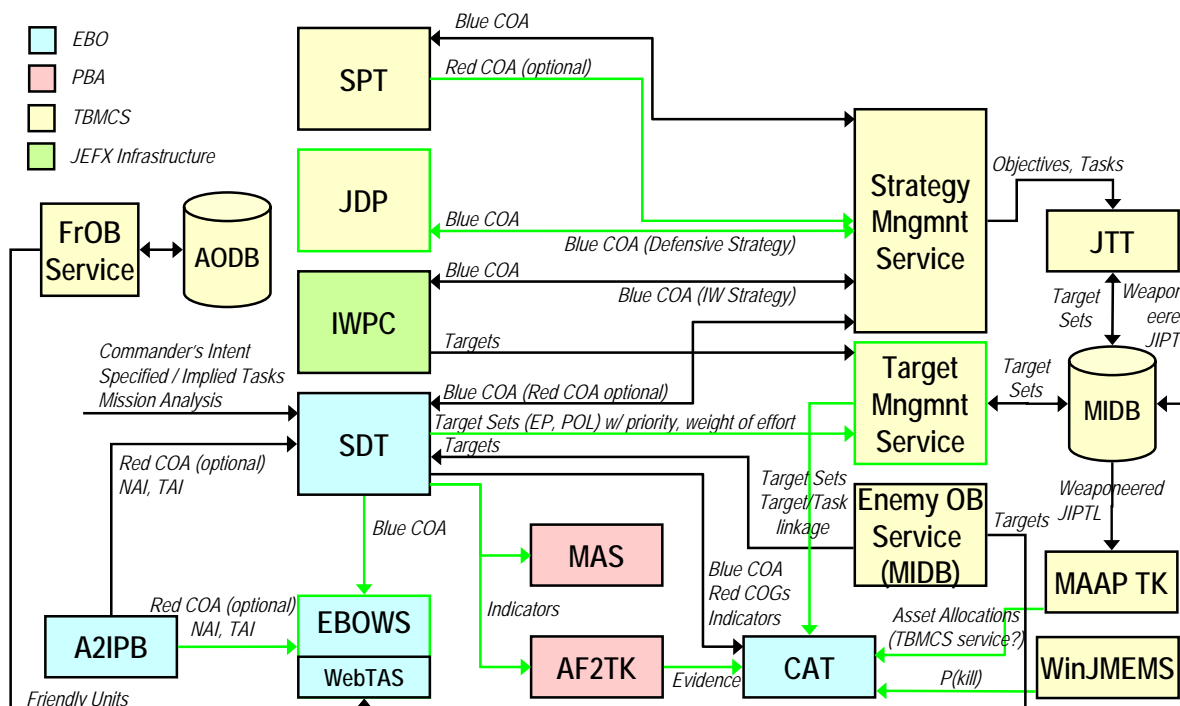
Figure 47. Final JEFX '04 SV-1 architecture.

JEFX itself consisted of four bi-monthly spirals starting with Spiral 1 in January 2004 and ending with Main Ex in July. The following sections give an account of the activities and insights from player observations recorded by BAE Systems for each spiral.

4.3.1 Spiral 1

Spiral 1 was held at Hanscom AFB in Bedford, MA. This spiral was actually more of a technology showcase for the accepted initiative tools, with the 8th AF players and various assessors present to get a preview of the tools and provide early feedback. BAE Systems had actually already visited 8th AF with AFRL to meet the lead JEFX player, Major David Dietrich, and the Strategy Chief, Lt. Col. Mark Matsushima. During that visit we briefed and demonstrated the full range of capabilities of SDT and CAT. We encountered an interesting contrast in reactions, from open-minded interest on the part of the JEFX lead to strong skepticism in the utility of specialized strategy tools from the Strat Chief. This varying degree of acceptance of EBO concepts turned out to be indicative of the overall reactions of the strategy team players during the main spirals.

During Spiral 1 BAE Systems successfully demonstrated SDT 1.5, including the newly developed delegation-based collaboration capability and undesired effects authoring. We worked with the technical assessor, Rick Abbiati, to answer questions about SDT and OAT capabilities, and held demonstrations and meetings with the operational assessors and end-users from 8th AF. During sidebars with some of the players, it became clear that even though SDT was the only strategy tool appearing in an initiative, it would be competing for attention with two JEFX infrastructure strategy tools: General Dynamics' IWPC and Accenture's SPT (part of TBMCS 1.1.3). Both tools exhibited significant overlap with SDT's COA authoring capability (in fact SPT was based in part on earlier versions of SDT and ISX's Effector). Despite the overlap, each of the tools included unique distinguishing features to be employed during JEFX. Only SDT provided COG model authoring and integration with effects-based targeting and COG analysis tools. SPT's discriminator was an automatic briefing and report generation capability, exporting to MS Office strategy documents. IWPC included special analysis and visualization for IO-related planning. Since AFC2ISRC had not committed to any one existing tool or a list of requirements for the future AOC weapons system strategy tool, JEFX '04 would prove to be the stage where all three tools vied for a chance for permanent AOC adoption. In the end, the assessors concluded that a single strategy tool with capabilities from all three was needed.

After Spiral 1, on Jan 27-29 we attended the Spiral 2 Prep event, in which 8th AF personnel worked through an effects-based COA development process for a Major Combat Operation scenario. Due to the limited training time, the technologists shadowed the players to develop COAs in the tools rather than having the users directly use the tools. Before leaving, we installed SDT 1.5.1 and OAT 0.2 on an AFRL laptop temporarily on loan to 8th AF so that users could further explore the tools before Spiral 2. A draft user manual, developed by our technical writer, was also delivered.

We attended two subsequent JEFX meetings prior to Spiral 2:

1) Spiral 2 Major Scenario Event List (MSEL) meeting – Chris White attended this meeting to present simulation inject requirements and data requirements from the EBO initiative's point of view. Chris also was able to help the C2TIG visualize EP and POL data within the SDT Target Set Tool, which aided them in data preparation for JEFX '04.

2) Spiral 2 minus 30 – Chris White again attended this event, supported by Lee Wagenhals of GMU. Day 1 covered general focus area training of EBO and PBA. Day 2 Chris and Lee held two 4-hour training sessions for two separate user groups. The training covered mainly COA development, plus adversary modeling and analysis. Dr. Wagenhals

delivered a one-hour tutorial on causal modeling. Day 3 was a "practicum" in which the AOC strategy team worked within SDT to perform mission analysis and COA development.

BAE Systems received many end-user suggestions during Spiral 2 minus 30, which were entered in SDT's bug/feature tracking database. One of the critical issues we encountered was the reluctance of the 8th AF users to build the kinds of detailed causal models featured in COG Articulator and OAT. Even the level of detail provided in SDT seemed to be more than they typically were willing to enter, preferring to stick with simple names, structure and rough timing. To mitigate this perceived complexity, we decided to emphasize during tool training that not all fields are required in SDT; users are free to enter as little or as much information as time and resources permit, starting from simply the plan element names and plan structure.

4.3.2   Spiral 2

Preparation

Just prior to Spiral 2 in late March, we worked on several enhancements requested by 8th AF during Spiral 2 Prep/-30 events. All of these enhancements also improved interoperability of SDT plans with SMS and SPT:

1. We added a "planning level" to objectives, effects, and tasks as a choicebox with choices of "Strategic," "Operational," and "Tactical." This would be used by planners more familiar with the Strategy to Task planning approach.

2. We added a "catalog ID" field to all plan element types except the top-level. This is a short string uniquely identifying the plan element that appears in parentheses in the Tree View.

3. We added a second echelon level for Mission Analysis and Commander's Intent screens covering the JFC level. A small arrow next to the MA and CI buttons allows the user to invoke the dialog with the data for the chosen echelon.

We also updated our software process documents to meet the standards of the JEFX MOA documents, including the System Design Document, SW Requirements Spec, System Test Plan, etc. These documents were delivered to AFRL in early March to meet the Spiral 2 MOA deadline. We also upgraded our security accreditation documentation to reflect the switch from Sun's J2EE app server to Apache's JBOSS.

For Spiral 2, we incurred a "functionality freeze" two weeks prior to setup, and a "bug fix freeze" one week prior, leaving a week to deal with critical software bugs or release problems. This led to a relatively smooth on-site setup at Nellis AFB during the week of March 22, despite stringent security conditions that made installation, testing, and patching difficult. The main problems encountered on-site were:

1. Lack of links data in the MIDB, which was rectified by the end of the week (however, the link data was too sparse to support Endstate TSA or link-based option generation).

2. Delay in TBMCS's setup of SMS, which was not ready till a day before training began, March 29. Despite this delay we were able to verify our SMS translators in time for execution.

3. The collaboration server would not run. During training week we discovered that this was due to a port conflict with JBOSS's naming server. Changing the JBOSS port to a new number worked around the problem. Unfortunately, during training, the collaboration server dropped out twice due either to network droppages or user overload. These scalability problems became moot when it became clear that users preferred a real-time parallel collaboration to our original delegation-based approach.

Training

Just prior to Spiral 2, BAE Systems developed extensive tutorial slides on different functional capabilities and planning process steps to be delivered as tool training for the EBO portion of the AFRL initiative. EBO Training was split into two sessions, the first covering Mission Analysis and COA development, the second covering COA analysis and collaboration. Day 1 covered the first session, with about twelve 8th AF users, several ops assessors, and a few other observers. Overall, the training went very well. There was some confusion at the beginning due to a few discrepancies in machines that either weren't set up with SDT or weren't installed correctly, as well as problems with the Information Workspace (IWS) screen-sharing and audio broadcasting software. We covered all the SDT plan elements, their attributes, and the major SDT views in the first half. The 8th AF users seemed to be keeping up well during the hands-on activities and asked insightful questions. Later, we covered mission and strategy templates and targeting tools. Dr. McCrabb described his counter-terrorism template.

After training, technical assessment went very well, with Rick Abiatti as our assessor again. We turned several capability assessment criteria ratings from "Partially Satistied" into "Satisfied," including analysis of COG models in CAT, and viewing and querying MIDB data. Endstate TSA and Option Generation were deferred to Spiral 3 since we didn't have adequate links data. Some items were deferred till Accenture's SMS was operational. Rick's main comment was that we need to expand the list of capabilities under SDT's COA authoring, since there is currently just one capability covering all the plan editors. A few other items were marked for expansion, to be done after Spiral 2.

Experiment Execution

The first day was spent performing mission analysis. Dr. McCrabb described this as a massive "cut and paste" job, gleaning content from existing guidance documents. The Strat Chief assigned various users to work on a particular SDT tabbed pane. The Judge Advocate General (JAG) worked on ROEs, while others worked on Resources, Facts, Assumptions, Implied Tasks, etc. The Strat Chief maintained the master plan on his PC, and the Info Workspace (IWS) collaborative tools allowed others to view his SDT window on their screen and send chat messages to each other. Since they could not grab control of his SDT application in IWS they had to send him chat messages with their contributions and he would cut and paste them into the tool. Aside from the lack of embedded collaboration, SDT performed adequately on this task. Accenture's SPT was used at the end of the day to generate a Mission Analysis briefing, but it filled in only about 20% of the Powerpoint due to a combination of SMS translator bugs in SDT and lack of SPT support for many of the needed fields. Because SPT supports only one statement per tab, the SDT MA text was mashed together so the users had to manually re-bulletize it. The strat Chief was not happy with the result since his team had to do a lot of follow-up typing and reformatting work in Powerpoint.

The second day was mainly COA development in SDT. The users split into two teams to develop two parallel COAs, one aimed at simultaneous direct attacks on Califon and the Nevidah terrorist movement, the other using indirect info-ops-based methods. The 8th AF's designated "SME" for SDT, Major Stew Greathouse, showed a keen interest in the COG Articulator and began using it to build an enemy model of the terrorist network and their ties to the Califon government. He had a network of 20+ red nodes by the end of the day and was coordinating with the ISR division to validate the linkages and persistence (workaround) durations. Unlike most of the other 8th AF users, who simply wanted to minimize their data entry, he had a penchant for using new tools and showed a natural intuition for this type of modeling. The two COA teams only made it down to the Effect level (operational level) by the end of the day because it took awhile for them to adjust to SDT's plan representation and decide on the appropriate objectives and effects. We had to push back on the tendency of some users to revert back to a strategy-to-task method of planning, which was very strongly engrained in their training and experience. Side SDT activities included joint demos with Accenture to General Santorelli and General Bordelon (JFACC) of how SDT and SPT were being used. The demos went well and both tools were favorably received.

On the third day the users continued to refine their COAs by adding timing information and in some parts, going all the way to causal linkages and tasks. One user was asked to enter the enemy's COAs: most likely and most dangerous. She did this only to the objective level, essentially paraphrasing in SDT what was provided already in Intel briefings. Throughout the day, the users were distracted by other side tasks as well. They spent two hours in an auditorium away from the tools doing COA wargaming. Dr. McCrabb played Califon (his strategy was to think what Saddam Hussein would do and do the opposite), and Cmdr. Thompson played the terrorists. They agreed they would eventually record the results in the COG Articulator, but no one got around to that - they seemed to assume that Maj. Greathouse had assumed sole responsibility for COG/CAT work. Unfortunately, he was frustrated much of the afternoon by crashes and bugs during COG articulator exports to CAT. Most of these were due to a mistake in the CAT installation process in which full access permissions were not granted on the CAT installation folder. After we fixed this problem, he was able to import one team's COA as a blue causal structure, place it alongside, the red model, and link blue effects as interventions against red COG nodes. After much experimentation with AND vs OR relationships he was able to show a significant impact on the terrorists' ability to project power due to blue interventions. Having to switch between the COG and CAT environment was cumbersome, as some changes were not transferring over and it would sometimes lay out nodes in ways he didn't like. Maj. Greathouse gave these tools poor marks for stability but good marks for usefulness.

In contrast, the Strat Chief was still pointedly skeptical of the utility of this "JWAC style" analysis in the strategy cell, though Major Dietrich, the 8th AF chief and Cmdr Thompson were supportive. The Strat Chief, because of the extreme pressure from the JFACC to produce deliverables for downstream AOC groups, treated JEFX more as an exercise with the goal of winning the war than as an experiment to evaluate new tools.

Toward the end of the day, Accenture helped the COA teams push their translated SDT plans into a Powerpoint briefing, but the results were again very thin and the teams ended up filling in a lot of content and formatting manually, and also including their own screenshots of SDT views. The Accenture briefing generators used only pure text and tables. One major lesson learned during this spiral was that the 8th AF wanted a fully collaborative tool, along the lines of BAE Systems' APS tool from Active Templates. On Day 2, one of the teams got so frustrated with

IWS' slowness and lack of ability to share apps that they switched to IWPC. IWPC field-locking and update sharing similar to APS, so the users could work together on the same parts of the plan simultaneously. When they were done in IWPC they would bring up SDT and re-create the same structure, then copy and paste content into SDT. So IWPC exceeded both SDT and SPT in collaboration support, though perhaps not in GUI ease of use and EBO support. Tim Autry, the IWPC trainer, said that users had been reluctant to use the IWPC analysis tools like IWCAT and COAST during IWPC training, which jibed with the users' general reluctance to use COG Articulator and CAT (except for Maj. Greathouse).

Meanwhile, SDT, CAT, and SPT continually expanded their presence on the JEFX network. In addition to eight PCs used by the JFACC senior mentors (several of whom were retired generals), it was installed on three active generals' laptops, two assessor PCs, all eight controller PCs, and all Guidance-Apportionment-Targeting, Strategy, and Ops Assessment PCs. The next day BAE Systems and Accenture reprised their SDT-SPT demo for Gen. Carlson, an active three-star general.

Throughout Spiral 2, BAE Systems collected pages of feedback and notes from the users. We wrote up an extensive bug/feature list for the next SDT team meeting. One of the most critical bugs was that DCOAD would not start due to a Java registry error. Eventually the DCOAD team was able to produce a workaround for the bug, but it was not ready to deploy until after Spiral 2. The problem was not a showstopper for JEFX, since A2IPB data could be viewed via Internet Explorer, but the web-based A2IPB view was not as user-friendly and informative as the DCOAD map view, and this failure was seen as hampering a high-level initiative interface between PBA and EBO, thus fixing the interface became a high priority task for Spiral 3.

4.3.3 Spiral 3

Preparation

Soon after BAE Systems returned from Spiral 2, we held a Configuration Control Board meeting to discuss the enhancement suggestions and bugs discovered by the players. We prioritized the list and assigned high priority features and bugs to appropriate developers. Because there was initially little work scheduled on major new views other than the Timeline Editor, we allocated the bulk of our labor between Spiral 2 and 3 to addressing these changes. We also briefed AFRL on three possible lines of development for new SDT capabilities aimed at addressing the most urgent needs of the JEFX players:

1. MS Office Integration for strategy briefings and documents, to shore up SDT's support for final production-stage activities, making it more attractive for AOC transition

2. New views for operational assessment and COA comparison, for better support for JAEP and JATO assessment processes

3. Real-time collaboration and database persistence to allow simultaneous updates to the same plan in parallel

AFRL approved tasks 2 and 3, reasoning that Accenture's SPT already supports the first capability, while the other tasks would address important operational technology gaps.

Training

The two days of EBO training went smoothly. Chris White and Ron James covered the refresher training with Brian Shimkin and Nick Pioch helping with questions, then Ron and Nick covered the second day of training for new people, with Chris and Brian helping. The users seemed happy with the changes. Since the SDT team was still working on real-time collaboration and the delegation-based collaboration was shown to be insufficiently scalable at Spiral 2, we presented a workaround method for collaboration using Microsoft NetMeeting to allow a user to share and grab control of another user's SDT window. The assessors were very interested in OAT and our PBA/EBO concept of employment slides.

Experiment Execution

Thursday was dedicated mainly to Mission Analysis. It started out slowly as the players figured out what they wanted to do and held numerous breakout meetings, but accelerated toward the end of the day. Accenture was working hard testing the SDT translators on the Spiral 3 plans built by 8th AF at Barksdale, and on the Spiral 2 counter-terrorist plans. Leo, the Strategy Doctrine lead, wanted them to get the Spiral 2 plans imported into JTT. They were having problems at first doing this because they had not regression tested this back home. They suspected the problems were due to the lack of the new plan schema content. It turned out that some of the guidelines hadn't been followed, like setting origin dates and using lower-case ".xml" in the plan name. They eventually got the plan over to SMS, into SPT, and from there into JTT. The Accenture team had to manually massage the SPT output to get it into JTT. Nevertheless, it was impressive to finally see the SDT plan inside the JTT strategy screen.

The main problems with SDT's SMS interface were slowness and memory intensiveness. The Spiral 3 plans were over 300 nodes, so the translators ran over 15 minutes before getting very slow due to a memory ceiling. Users had to resort to breaking the plan into phases, exporting each phase separately in about 15 minutes each, then merging them together in SPT. The SPT export was much faster since it was Java-based. Obviously, this caused a great deal of frustration, leading to complaints about having to use multiple strategy tools for the same purpose. BAE Systems and Accenture agreed to work together after Spiral 3 to optimize the translators before Main Ex.

In the morning, BAE Systems assisted Lt. Frankie Green, the ISR ACF liaison in the procedure for logging into A2IPB and subscribe to IPB data. This comes in as an email with the XML in the body. Because A2IPB doesn't send it as an attachment the user must manually save it, edit out the email header lines, and then import it into DCOAD. Other than that, it worked without a hitch. DCOAD's map view provided more details and more responsive navigation than A2IPB's web interface (the full A2IPB client was only cleared to be used in the ISR Division's Top Secret lab). A few suggestions for improvements included optionally adding map labels on the items and allowing toggling of the three types of data (points, regions, routes). The IPB data was not well fleshed out at this point. Named Areas of Interest (NAIs) and Target Areas of Interest (TAIs) were all very small dots, not regions. Three red COAs were entered with minimal description.

BAE Systems was invited to observe the Mission Analysis brainstorming session in the Red Flag building next door to the CAOC. It was interesting to hear the players' discussions and thought processes, despite that 80% of the material had been pre-prepared. Capt. Colazzo was the scribe, capturing their inputs in Powerpoint. At the end of the day she began entering the

items into SDT's MA screens. The next day they finished MA and begin COA development. They had a whole week to do the process this time instead of just the four days allowed during Spiral 2. However, they had more overall tasks since they also had to do operational assessment on the Spiral 2 scenario's execution, even as they finished the Spiral 3 scenario plans. Ideally, they would have been using OAT during this stage, but 8th AF had unilaterally decided to use JEFX '04 to test an internally-developed spreadsheet-based assessment tool called BOAST-COAST.

For the last few days of Spiral 3 execution, Brian Shimkin remained as sole SDT on-site representative. Brian, who was the lead developer of COG Articulator and recent OAT interface enhancements, primarily worked one-on-one with Maj. Greathouse, who continued to refine and analyze his enemy COG models using those tools. Thanks to many bug fixes made prior to Spiral 3, Maj. Greathouse commended the team for providing a much more stable environment, though there were still occasional glitches and lost data. In most cases, Brian was able to propose workarounds and manually retrieve lost model data. Maj. Greathouse exhibited remarkable creativity and resourcefulness as he became more intimately familiar with OAT's causal modeling semantics. Even though the ability to save and reload AND groups was still pending in COG Articulator, he discovered a way to replicate AND group behavior using only G and H values (causal strength and inverse strength). By the end of Spiral 3 he had built a complex "system of systems" model, initially building distinct subgraphs for Califon's military, leadership, infrastructure, and economy, then linking these subsystems together by identifying cross-COG points of influence. The validity of Maj. Greathouse's model assumptions were called into question by some strategy players, but he had been coordinating with ISR for IPB source documents and answers to model questions on these enemy systems. At the end of the Spiral, Maj. Greathouse briefed his models and analysis results to the JFACC to show how they helped back up the team's choice of a particular COA.

One of the biggest problems that arose late during Spiral 3 was the slowness of the SDT-to-SMS translator, which was used heavily toward the end in order to get the plan into SPT to produce automated strategy briefings. While most translation bugs had been fixed, the slowness was exacerbated during times of high network volume to the point that users sometimes had to resort to retyping plan data into SPT manually. When a wide variance in translation times was seen, Brian helped exonerate suspicion of SDT translation failure by performing a systematic series of tests during low network usage that showed translation time for the current plan consistently came out close to twenty minutes. Optimizing the translators would be a high-priority focus for Main Ex.

4.3.4   Main Ex

Preparation

The primary development focus between Spiral 3 and Main Ex was real-time collaborative planning. SDT's involvement in Main Ex would occur in two distinct phases. The last week of July, SDT would be used as it had before, to round out plans developed during Spiral 3. Also, a portion of the Ops Assessment team would evaluate OAT's utility for operational assessment of the Spiral 3 plan currently being executed that week. Then, the first week of August a smaller contingent of strategy players would move to a special trailer to work on *branch planning*. This involved applying a compressed planning process to address a contingency event specified by

the CFC. This smaller group would evaluate SDT's newly developed real-time collaboration capability.

We had a head start on requirements and design for collaboration prior to Spiral 3, but the bulk of the implementation and testing occurred the two months preceding Main Ex. To assist with the daunting task of multi-user load testing and to allow for parallel testing and development, BAE Systems hired a small team of consultants at Ajilon Consulting in Nashua, NH. We held an orientation meeting with them in late June and prepared a detailed Unit Test Plan covering the main collaborative planning functions and likely error conditions. Ajilon helped us discover a memory leak in the JBOSS application server causing the system to crash after moderate load and usage time. After attempts to streamline the JBOSS configuration failed, we resolved this problem by abandoning JBOSS and migrating to Exolab's OpenJMS system. The advantage of this system is that it is strictly limited to JMS messaging, and does not provide all the additional services included by default in a full J2EE application server. This greatly reduced the memory footprint of the JMS server. More importantly, it operates by default in a mode in which all messages are stored into an internal database, rather than using in-memory storage like JBOSS. Therefore, the memory used by the JMS server is more or less constant at a low level throughout the entire session, unlike JBOSS, which would accumulate continuously until overflowing. Fortunately, the impact on the rest of the system was very minor, since both servers adhere to the JMS standard, and we had not utilized any JBOSS-specific capabilities. Most required changes were in the SDT configuration files and installer creation process.

We continued to fix bugs in various areas of the collaboration API as they were reported by Ajilon Consulting, and provided at least one system update per day to Ajilon. Fixes and regression tests focused first on functional capabilities, then on error-checking, then on scalability, and finally on robustness. After mid-July, BAE Systems continued to test and debug collaboration internally up to the branch planning setup trip on July 28. At this point, several minor bugs still remained, so we created a document listing the known problems and suggested workarounds. We also defined high-level guidelines for proper use of real-time collaboration, such as avoiding use of Timeline or Structural Editors while collaborating, and avoiding frequently working on the same areas of the plan. Also, due to some problems propagating updates to Phase nodes, we suggested working on the high-level plan structure down to phases offline, then uploading the plan to the database for collaboration.

Setup and Training

At the request of AFRL, BAE Systems split its team to provide simultaneous on-site support at two remote collaborating JEFX sites. Chris White traveled to Hurlburt AFB in Pensacola, FL to setup the tools and provide training to an Army contingent building plans for the ground forces component in SDT. Chris stayed at Hurlburt to provide on-site technical support for the entire duration of Main Ex. The rest of the team alternated traveling to Nellis AFB to support the 8th AF players, with Brian Shimkin and Ron James attending the first week, Brad Pielech attending the second week (branch planning) and Nick Pioch attending the entire duration.

For completeness, the following is an unabridged copy of of BAE Systems' software engineering team's detailed notes taken during the JEFX 04 Main Experiment.

Friday July 23

BAE Systems completed installation of all four systems in Building 9402 (the trailer to be used for branch planning). Each system has the collaboration client and server installed. We did not do any collaboration configuration, as there was interest in some of the new players in using those systems prior to branch planning to better understand SDT, and we did not want to risk them using collaboration before it is officially ready. We made two exceptions, installing collaboration for Maj. Greathouse and Lt. Col Hunerwadel, as they were interested in collaborating while "pre-branch planning." They were aware that they would be using an early version and there was a need to tread lightly.

Buster McCrabb gave remedial EBO and tools training for some of the players. When the issue of translator performance came up, BAE Systems responded that in Spiral 3, translator upload was 20-45 minutes, but now at Main Ex the same plan uploaded the previous night in 11 seconds. Buster was impressed, but no one else seemed interested, even those who were most affected by the slow performance in Spiral 3.

Ron James (the Endstate Program Manager), gave a demo for General James, chair of the EBO working group. Col. Cassarino, from the Electronic Systems Center, who works with Gen. James, asked a lot of good questions, both detailed and from a big picture perspective. Overall, he seemed to resonate with SDT's capabilities. Buster spoke about SDT and did an excellent job, portraying it fairly and in a very positive light. After lunch, Col Cassarino had returned for an Accenture demonstration of SPT. Ron described the SDT's Endstate analysis capabilities further with Col Cassarino. One of the Col.'s key questions was how SDT took the planners from effects-based planning to effects-based execution.

Brian worked with Lt. Kelly Friesen of 8th AF on the COG articulator, who was slowly coming up to speed on COG modeling, but had had very little tool feedback from other more experienced users. The majority of the support that day surrounded the theory and methodology of adversary modeling.

After the new SDT-SMS translators were loaded, Accenture and BAE Systems successfully verified a two page list of SDT-SMS data mappings. The next day, Capt. Colazzo approached BAE Systems to verify that SDT would not lose data on the upload. We told her about the successful test results and assured her that it would not, but she seemed dubious. The players have spent the past couple of days sending plans from SDT to SPT, back to SDT, and back to SPT. This has flushed out some software errors.

Experiment Execution

Monday July 27

The tools received light use Monday, with the exception of Lt. Friesen, who was still wrapped up in COG modeling. However, Brian overheard him working with another ops assessment player (Lt. Benjamin), relating the COG model information to that of BOAST COAST (an Excel-based spreadsheet tool developed by 8th AF for operational assessment). Brian spoke with Lt. Benjamin at length about BOAST COAST. Interestingly, she didn't feel it was a real "tool," and conveyed a wish to have a more traditional software GUI approach to

make it a real tool. She indicated that she was spending a great deal of time in BOAST COAST performing tedious reformatting changes. Maj. Greathouse continued to preach the "gospel" of COG modeling to anyone within earshot.

Buster asked BAE Systems to re-verify SDT availability in Bldg. 9402, and to set up the projector. He spent several hours with Gen Croker in 9402 while Gen Croker explored SDT. Buster indicated that General Croker "loved" the tool, wishing only that it had spell checking, underline capability, and text copy via highlight.

Jake Tweedy, the lead Ops Assessor, later pinged Brian again about status of the SDT collaboration build in 9402. Brian repeated his earlier answer of Thursday July 30. Jake decided that it needed to be available first thing Thursday morning, and had asked Brian to expedite it if possible. Brian spoke to the FDO (JEFX help desk) today, but they were busy with other requests, so he resolved to touch base with them again the next day.

Tuesday July 28

During rare spare moments, Brian (at Nellis) and Chris (at Hulrburt) had been periodically phoning in to perform "cross-continental" collaboration tests on SDT. Tuesday they were able to successfully connect to a database plan and enter data into the same plan in real-time. The Strat Chief was pleased to hear about this Hurlburt/Nellis collaboration capability, and suggested getting input from higher headquarters as to its use during next week's branch planning. He was suggesting that there is increased value for other JEFX focus areas, i.e., net-centricity.

Today no further issues arose regarding the SDT-to-SMS translators. A reverse translator issue was being addressed by Accenture via a template modification. Brian asked if there was anything BAE Systems could do to help, and they answered "not at the moment," but did put in a request for Brad Pielech to possibly code around a discrepancy discovered in SPT.

Rick Abbiati, the lead technical assessor, announced he would like to do a tech assessment of DCOAD. Brian had at this point successfully got DCOAD to launch and load A2IPB data. Collaboration tech assessment was tentatively scheduled for Friday. Rick also verbally verified with Brian how the DCOAD/A2IPB input took place (importing email attachments).

Thursday, July 31

The demo to ESC's General Johnson went well. Mark Pronobis of AFRL briefed the PBA portions and Buster briefed and demonstrated SDT, COG Articulator, and OAT, with prepopulated data. He also showed SPT and the briefing composer results. He was equally positive about the AFRL and ESC tools and emphasized the cross-tool interfaces and SDT's collaboration.

(SDT developer) Brad Pielech and Nick Pioch got a summary from Brian of the week's events before he left for home. The complaints about SPT-to-SDT translators had died down, and Accenture was working to fix numerous small bugs on their end as well as in their briefing composer.   One problem was that SDT was using more digits in its long UIDs than before. Despite satisfying the SMS spec, the longer IDs caused problems with the SPT's Strategy Briefing Composer. Accenture had been working in Bldg. 9402 to test fixes and coordinate patches with their Reston, VA team.

Brian had already installed SDT the latest collaboration patch onto the Building 9402 machines and said the performance seemed to be much better than last week's engineering build. Brad and (AFRL EBO POC) Geri Rogers ran more stringent tests during the afternoon and also had positive results. They did not get a single failure over an hour of testing. Nick prepared a handout of guidelines and caveats with collaboration that were aimed at preventing users from straying too far afield.

General Croker worked on a JFC-level branch plan this week after an SDT tutorial from Buster. He was able to build this in eight hours, working with a few other senior mentors and staff. He sat with Brad and Nick for about 30 minutes, describing some "nit-picks" about the GUI, including minor bugs in the Timeline and Properties editor. Overall he was impressed with the tool.

Meanwhile, Maj. Greathouse and Lt. Col. Hunerwadel were building mid-size COG models in preparation for the branch planning. They described this as more of a "wargaming" model than red COG model because it had a significant number of multi-level blue nodes in it intervening against an equivalent number of red nodes. Lt. Col. Hunerwadel had been converted to the "gospel" of COG modeling. Maj. Greathouse asked him if he ever would have thought of using a tool like COG Articulator? Answer: no. Now that he's seen it, would he ever want to plan again without it? Answer: No.

However, today a minor bug arose in their model preventing the nodes from showing up. The log file ended with a StackOverflowException. After some analysis of the COG model file, the error was diagnosed to be due to a cycle in the model graph, but the cycle was not from the user but due to multiple nodes having the same ID so the links that originally were on two different nodes got merged together, creating an artificial cycle. It was likely that the duplicate node IDs arose from a copy and paste operation. The ID generation for COG models needs to be upgraded to the same as that of SDT to avoid duplication. Nick manually repaired the model and asked them to avoid copy and paste for now.

Lt. Friesen, who had been working with the old Spiral 3 COG models, was finally asked to enter evidence in OAT for assessment today by Maj. Greathouse. He gave him an Excel spreadsheet of about twenty red COG nodes on which to enter probabilities. The probabilities in the spreadsheet were from blue's perspective, that is, 1.0 means 100% destroyed. So, because the red model represents probabilities as likelihood of enemy nodes being active, Kelly needed to enter the inverse probability, (1-P). Nick helped Kelly do this for a couple examples, but the Spiral 3 red model with numerous blue interventions is so large that OAT completely maxed out CPU to the point where he couldn't get the Evidence Manager to respond. Later, we figured out that you can do most of the evidence operations while analysis is paused, except for actually bringing up the manager from the View menu. Even so, analysis under Effects DBN on such a huge model was frustratingly slow. Nick recommended to Kelly to either reduce the size of the model, or move to pure blue COA evidence analysis.

Friday July 30

COG ARTICULATOR

No further bugs in this area to report. Maj. Greathouse clarified that one of his red "models" is really not a causal model but a decision tree. The top red node represents a high-level decision the enemy will be faced with, and the links are named "yes" and "no," leading to yet another

decision. Some links were labeled "if" and "then." Often the decisions alternated between red and blue sides to convey a wargaming flavor. This shows the flexibility of COG Articulator to support yet another type of useful activity to prepare for COA development.

<u>ENDSTATE TRAINING</u>

In the morning Nick worked with Maj. Greathouse to train him on the Endstate tools, as he was eager to see if they could support the branch planning. Nick showed him the map tool and map legend, on which he immediately asked to see the Califon V Corps ground units. Nick tried doing this with the legend, but there were so many disparate category codes applicable for that unit that it was difficult to enter a reasonable number of discrete ranges that would include all of them without including other targets outside V Corps. Instead Nick tried using the query tool, and after picking the most likely-sounding category code, he could scroll down the result set to see all the V Corps targets listed alphabetically. After selecting and showing the names, they all showed up in the Sacramento area, apparently their home base. Later that day, Chris White independently confirmed this result from Hurlburt, which unfortunately meant that the controllers had not entered proper MIDB or IPB locations for V Corps, making it very hard for the Army to do its planning.

Maj. Greathouse directed Nick to skip to the most automated Endstate use case. This would be Target Systems Analysis, which predicts outages over time on a set of indirect targets given a user-specified set of direct targets and strike times. When Nick mentioned that Endstate currently only handles EP and POL networks, Maj. Greathouse replied that that would not be useful for branch planning, since he needed to evaluate targets to delay or isolate V Corps from reaching a Nevidah destination. Ironically, this would have been a perfect problem for Workarounds Reasoner, the first target system tool Endstate had integrated with, but one that became obsolete once we moved to the OpenMap map tool. Ron James would be flying in again mid Monday to provide technical support for Endstate.

<u>COLLABORATION</u>

Fri morning Brad, Geri, and Chris performed a successful cross-continental collaboration test (Nellis to Hurlburt), with all participants actively editing the plan. The updates occurred almost instantaneously, with no apparent lag compared to local collaboration. Later, we notified Gen Croker who suggested that Nellis and Hurlburt collaborate on the same plan. In fact, Hurlburt had already finished MA and begun COA Dev, since they started immediately after the WARNORD came out Thurs. 8th AF and Gen Carlson apparently were not happy about this, and eventually both Nellis and Hurlburt were told to put all planning on hold till Monday morning and work together in parallel. Chris reported later that this greatly frustrated the Army who had seen a window of opportunity Thursday to stop the enemy and waiting till Monday to plan would be too late.

At the end of the day, Brad and Nick gave an informal tutorial of collaborative planning to the lead branch planners, Maj Dietrich, Dale Gray, Lt. Col Andy Carter, and Lt. Col. JP Hunerwadel. They seemed very pleased with both MA and the COA-level collaboration, despite somewhat slower response rates probably due to a network reconfiguration by IM (JEFX Information Management) at that late hour.

Everyone in the Strategy Division was given the weekend off except the most senior members, but they said no SDT or SPT support was expected. Unfortunately, the Hurlburt AFB component was not so lucky, as Chris felt it would be important to support Cmdr Pettit as he

scrambled to meet the frequently shifting tasking coming from Nellis higher headquarters. Branch planning was to begin at 7 AM Monday, so Brad and Nick planned to be there on time in 9402 to help kick it off.

<u>EVIDENCE ASSESSMENT IN OAT</u>

Recall that Lt Friesen had been tasked by Maj. Greathouse to work with OAT to enter a list of canned evidence probabilities on red COG nodes in the model. Lt. Friesen was obviously less comfortable with modeling than Maj. Greathouse and seemed to be spinning his wheels while Maj. Greathouse was busy working on other tasks, so Nick offered to provide help.

In the afternoon Nick worked almost exclusively with Lt. Friesen after learning that he would be leaving the next day. He was the only one tasked to try to analyze evidence in OAT, so we wanted to ensure an earnest attempt by an end-user to work on this had been observed by the assessors. Lt. Friesen had been frustrated the days prior to that because of sluggish performance in OAT making it difficult to navigate the evidence GUI while analysis was going on. We found that many of the clicks could be done while analysis was paused. This allowed him to work faster, but we were both still having a difficult time getting meaningful non-evidence results let alone evidence results. After some investigation, this turned out to be because of the lack of support for G-values in Effects DBN. Maj. Greathouse had put these in many places to simulate AND groups with G-values of -1. In Effects DBN mode, the "AND" nodes were always coming out true, even if one of the causes was false. Nick was able to fix this by adding an explicit AND group in OAT using the Probabilities table.

Now that we had established a degree of model correctness, I asked Lt. Friesen what goal had been given to him for the analysis (that is, on what nodes are we interested in the impact of evidence), and he said that hadn't been discussed. I pressed Maj. Greathouse for something specific to look for, and he suggested reducing the scope to a single piece of evidence on a key EP node. The evidence should be that the red node is operational (Prob = 1), despite our having bombed it on Days 1,3,5, and 6. Maj. Greathouse then wanted to see the impact of evidence on the downstream EP node and on "Zone 2 SAM Engagements," since Zone 2 depends on this EP node.

At this point Nick called in Rick Abbiati, since he had expressed interest in seeing OAT evidence analysis in an operational setting. We entered the evidence as 1.0 for twelve days, and after waiting awhile for convergence, we got a somewhat unintuitive result for the node on which we entered the evidence. It did show the graph for the observable EP node as 1.0 for the first twelve days, but then it dropped straight to 0, which is what it would have been if the interventions had succeeded. But because the last intervention on Day 6 should have had no delay, OAT should have concluded that all of the interventions failed based on the evidence that the EP node was working on Day 6. Thus OAT seemed to be saying that despite the evidence that the direct effect was not achieved when it should have been, after that observation period expired, the effect was assumed to have suddenly been magically achieved. Nick didn't have time to verify the timing and probability inputs all the way down so it was not certain there was a bug, but we resolved to recreate this scenario on a much smaller scale later. However, we did see some expected results as well, such as the original intervention on Day 1 being reduced from high probability to very low, and each subsequent intervention also being reduced, though less as the date increased. The effect on the downstream EP node and Zone 2 SAMs also seemed to match expectations. On the whole, Rick seemed to be pleased despite the mixed results.

Note that some of the problems with evidence results may be due to Maj. Greathouse's intentional misuse of a single blue node as representing multiple separate real interventions, with the inhibitor links representing the different actions. Nick had warned him that OAT really wasn't designed for this, though acknowledged it would be a useful feature.

Monday, Aug 2

Branch Planning

Today was the start of branch planning using SDT's newly developed real-time collaboration. The branch team worked on Mission Analysis all morning, with three different users dividing up the SDT's Mission Analysis tabs to work collaboratively in parallel. Overall this worked very well, much more smoothly than during Spiral 2 when they had to send all their inputs to the Strat Chief via a chat tool. A few minor bugs and limitations were found:

1. Clicking on the OK button multiple times when the initial lag was more than expected led to multiple check-in requests on the server. The second click caused an error dialog prompt, as if the user were trying to commit without having the lock.

2. One COA team lost some entries in the Implied Tasks tab, supposedly at a random time when someone else committed or changed something. Unlike most cases, reloading the plan did not retrieve the work, so the problem affected the database. This helped to drive home the need to save repeatedly to files.

3. Later when one user was talking through his tabs to the team while broadcasting the MA view through IWS, he would make updates to the MA tabs on-the-fly based on feedback from the audience, but the MA tab would dismiss after each change. They wanted to see these changes on their SDT screens as he made them, rather than sit there with the entire tab locked in the original read-only state. (This was a known limitation – the SDT team had only had time to implement tab-level locking in the MA screens, not field-level locking).

At 11 AM they exported the plan to SMS and generated a briefing. It took a couple tries to get this just right, due to stringent requirements in SMS regarding date origins and other fields. Then they briefed the MA results to Gen. Croker, who was playing the CFC. He directed the team to focus more on the experience with the tools more than the actual MA content. There were many minor suggestions on how to improve the collaboration as well as minor suggestions about the briefing content, but both Major Dietrich and Gen. Croker seemed very pleased with the collaboration support.

In the afternoon, they began COA development. They broke into two teams, COA1 (Maj. Dave Dietrich, Lt. Col Andy Carter, Lt Col JP Hunerwadel, et al) focused on close air support, and operational assets, and leading elements, while COA2 (Maj. Greathouse, Capt. Colazzo, Ret Maj Ed Wempe, and Lt. Frankie Green) focused on logistical support for the Army. They spent over an hour brainstorming separately on whiteboards. They sketched out ideas along categories of "Means," "Ends," "Risks," and "Effects." They also discussed together how best to utilize the skeleton JFC-level plan provided by Gen. Croker as a startpoint for the Air Component plan. Then around 2:30 each team began entering the COAs together collaboratively into the same SDT database plan. This worked well for about an hour, then they ran into some problems.

At that point, they started trying things that the BAE Systems team warned during training might cause problems. Lt Col Hunerwadel tried copying and pasting an Effect fragment, but only the root Effect node was copied. BAE Systems suggested closing and reopening the plan to check if it really updated the DB correctly. Maj Greathouse had also just moved to a new station, so he also tried re-downloading. Both systems seemed to hang at 80% complete. Numerous attempts to reload failed. Luckily, they had been saving fairly frequently to files, so not much work was lost. Each team re-uploaded the plan from a recent file, and then we tried walking them both through saving fragments as mission templates and inserting them to simulate copy paste. This seemed to work at first, till Maj Greathouse discovered that changing the inserted nodes from the template also caused the original nodes to change. Thus, the database plan really thought they were shortcuts with the same node ID. SDT's algorithm for assigning new IDs to pasted fragments would only have been invoked when uploading a plan to the database, not on insertion of mission templates, so if a user inserted a template just created from another node in the same plan, the IDs would still conflict. Later that night, Brad Pielech looked into a fix on the "insert template" operation that ensures unique IDs for each inserted plan node. Meanwhile, the users had to work around the problem by creating a plan full of just the templates, uploading it to generate unique IDs for the elements, then downloading it again and re-saving the templates with the new IDs. Then they could be inserted into the plan.

Even with this workaround, they still experienced plan loading problems. After over five minutes SDT would abort the operation, saying it could not load the plan. Brad checked the logs and believed this problem was due to a circularity that sometimes crops up in "object of Task." Another minor issue that was causing some complaints was the automatic prepending of "Copy of" in the "Catalog #" field when a node is pasted. It should only do this if the previous operation was a copy, not a cut.

Due to the above problems with reuse of fragments, the two COA teams decided to work offline for the rest of the day in order to finish the significant copy/paste/tailor operations required, then try collaboration again toward the end of the planning activity.

Maj. Greathouse enthusiastically demonstrated his adversary modeling work in COG Articulator, advocating broader use by the COA teams. He built yet another red model (this one a real causal model rather than a decision tree) which he briefed to the entire group. He recommended elaborating objectives by exporting red causal chains to SDT. Major Dietrich had not really seen this before and registered curiosity but was still outwardly skeptical, asking "how will this cut the planning time down?" This reflected a general tension at JEFX between a tendency to favor new tools and processes that reduced *task completion time* as opposed to tools that inject new activities in order to improve the *quality* of the planning products.

A2IPB IMPORT

Lt. Green said he was still having problems importing A2IPB data. The log file pointed to a problem parsing <COMINT> tags in the A2IPB file. He put me in touch with a William Poppert, a Zeltech A2IPB developer, who said Zeltech had been posting XML from a newer A2IPB version containing tags not expected by SDT's parser. His developer, Rod Lehman, emailed BAE Systems' DCOAD developer John Stucklen the new XML schema for A2IPB, so John could determine the extent of the version skew. John had already posted a new patch, though Chris White reported two other fields that caused problems. Meanwhile, Chris edited the A2IPB file manually to remove all unknown tags.

BAE Systems' Chris White, who had decided to stay at Hurlburt throughout Main Ex, periodically uploaded the CFLCC team's plan to the Nellis collaboration server. That way the 8th AF strategy team could browse it at their leisure. Likewise, Hurlburt's team could configure a machine to connect to the Nellis collaboration server and browse the AF plan. They agreed this was preferable to collaborating on the same plan across the two sites. Specific praise was given at the MA hotwash for this type of collaboration among the Land & Air components.

Tuesday Aug 3

On the whole today was a successful day for EBO. In the morning Maj. Dietrich directed the team to spend time in COG Articulator building COG/Target System models, since there was a step for this in Dr. McCrabb's process checklist. They worked on this collaboratively with several people working independently on red model fragment contributions, then one person putting them together. The ISR Tgt liaison user employed Nick Pioch as scribe since he wasn't comfortable with the tool yet. Lt Cols Hunerwadel and Carter seemed to get a good feel for the tool by the end of the activity. Even though not everyone grasped how to build true causal models for analysis in OAT, the important lesson is that the qualitative models they did build stimulated discussion about the enemy that helped prepare them for COA development.

Recent conversations with Dr. McCrabb and EBO Assessor Jake Tweedy confirmed that the assessors were seeing the benefit of COG Articulator by watching one COA team constantly refer back to the red model as they were building the COA. Gen. Croker also seemed very pleased with the activity during the midday hotwash. There were some comments from other more skeptical users about not understanding the "spider web" view [referring to COG Articulator] and preferring a table/spreadsheet view, but no arguments about the utility of the information. Some people suggested that this type of modeling should be done mainly by the ISR Div, but Dr. McCrabb said for EBO it should be a shared responsibility with Strat Div.

During lunch with Maj Greathouse, he presented some ideas for how to adapt SDT into a joint-level tool. He suggested we should allow arbitrary MA / CI windows for the different echelons and components, CFC, CFLCC, CFACC, etc. He suggested layering component-level plans on top of the original CFC plan, with pedigree information kept on each plan node as to which level or component worked on it. Then you could filter the plan to show just the level/component content you're interested in.

After lunch they resumed COA development. Brad's patch seemed to resolve the problems with unintended shortcuts from mission template imports. It took over an hour to get everyone resynchronized because each team had done a significant amount of offline work, and one team got impatient waiting for the other one to upload, so they continued to work offline without BAE Systems' knowledge. That meant we had to re-merge their new content with the uploaded plan. Loading plans takes 5-10 minutes, so we needed to enforce a hands-off policy while resynchronizing to minimize wasted time. After this, they had no problems for over two hours, making significant progress on both COAs. Toward the end of the day we coordinated with the Army on the timing of the tasks and effects.

At the very end one user's plan got slightly out of sync with the database, as compared to the neighboring SDT display. Also, there was a minor bug in transferring the plan to SMS in that the plan was so large some SMS IDs became too long for the server, so Brad agreed to work on a

patch that ensures they fit the desired range. This was not a huge issue, as they just wanted to be able to share the plan with IWPC. The next day they would need it for generating COA briefings.

Wednesday Aug 4

On the whole, it was another successful day for EBO, but some minor setbacks occurred, which fortunately did not seem to make a big impact on the positive impression of the EBO tools. By the time BAE Systems arrived, the two COA teams had started working independently on their COAs offline. It was taking too long to repost, merge, and re-download the plans collaboratively, so they continued to work this way the rest of the day. Only minor changes were left to do in the plan:

- MOEs/indicators

- COA descriptions/summaries

- Weight of effort (WOE), priority

Cmdr. Thompson made a suggestion that users enter time-phased WOE for each task, e.g. 20% first day, 40% day two, etc. BAE Systems reported that SDT only supports a single WOE, but that it could be updated and reposted daily. Dr. McCrabb suggested copies of tasks could be used, each with different WOEs and days. Also, BAE Systems suggested using the Planned Assessment tab's %-completion column to be interpreted as WOE instead.   Gen Croker seemed impressed with the flexibility of the tool to be so adaptable to such a spur of the moment requirement.

When the COAs were finished, we conferred via telecon with the Hurlburt Army team again to brief the air vs. land COA descriptions over IWS using the shared view of one another's SDT screens.   They mainly wanted to ensure a common understanding of the timing of each others' tasks and effects, expectations of enemy behavior, and intent of the COA. The Army group suggested a Deception COA in which blue holds off on air attacks in order to lure the enemy into a particular region; this was deemed as risky by the 8th AF in discussions afterward.

After this, they moved to COA wargaming, using COG Articulator at Maj Greathouse's request. He served as scribe for this effort. He exported the blue COA to COG Articulator, then started recording initial enemy actions leading to a chain of blue and red counter-reactions. Most blue actions were drawn as new blue nodes, but some were already in the plan. Here, he dragged the existing blue tasks down to the right spot. This made it clear what had already been planned for and what needed to be added. For the "grand finale" Maj. Greathouse hooked up an inhibitor from a blue box to a previous red box, highlighted the causal chain, and exported it back to SDT as an effect/task fragment. This use of COG Articulator greatly impressed Gen. Croker and Maj. Dietrich, who saw it as a major improvement over simply writing out wargaming results on the whiteboard. Note that this activity was performed collaboratively with Hurlburt over IWS shared screens and phone teleconference. One user was still skeptical - she filled out a tabular version of the wargaming in Excel to convey her preferred methods and displays. Dr. McCrabb commented that they should have picked their COA analysis criteria before they began wargaming so they could have used the wargaming to answer how each COA scored along the criteria.

In the afternoon, one COA team worked with the COA Comparison Matrix to tailor criteria and fill in scores for the two COAs. BAE Systems had to remind them that this was available in SDT, since it had been awhile since they had received tool training. They said it worked exactly as they wanted, and they were able to fill it out in under an hour. Of course COA1 won, since that was the team filling out the values, much to Maj. Greathouse's chagrin.

Meanwhile, Maj Greathouse worked within OAT to try to analyze the COAs probabilistically. Unfortunately, he encountered an export problem here. He could only export his COA, COA2. The full merged plan would not export any nodes at all. In COA2, many effect nodes were not connected up to their parent, so he only got a partial result for this. There wasn't much time to manually compensate for this, because both teams had not filled in many start/end times on the tasks, which was required for OAT analysis.

The plan also still had trouble fully translating over to SMS. After a lengthy upload time due to the slow network, the SMS server responded "Invalid Plan" with no information as to where the problem occurred. Debugging this can be like finding a needle in a haystack. Accenture was not available to help, having left in the early afternoon. When they arrived later, they investigated but could not find the cause of the problem. Turning to the players, we found out the only reason they wanted to do this export was to get the plan into IWPC to look at its synch matrix (timeline). We were able to do this using an earlier successfully exported plan, at least fulfilling the spirit of the activity's purpose if not with the exact final plan.

Thursday Aug 5

In the morning Nick worked with Maj Greathouse to see if they could at least get part of each COA exported to OAT. Exporting the entire plan would not send any nodes, with no error message in the log file. COA2 exported fine, so the problem was somewhere in COA1. We tried exporting each objective one by one and all of COA1 phase 1 objectives came across. So Maj. Greathouse created a partial plan made of Phase 1 from each COA and this exported okay. Some of the effect nodes were not linked up to their parent objectives (again, no apparent error messages) but not enough to significantly impact the results. Maj Greathouse generated graphs for both COAs and showed them to Gen Croker. COA1 peaked at 0.5 and stayed level for a few days, but the peak occurred earlier than COA2. COA2 peaked at 0.6 and lasted longer but peaked several days later. Drilling down to objectives showed this was because the attack on red's V Corps was scheduled earlier in COA1. Based on this, Gen Croker recommended selecting COA1 to get an earlier indication of success. Gen Croker was pleased with the OAT results and its temporal drilldown process, and asked Maj Greathouse for some more customized probability profile screenshots for the end briefing.

After this, they conferred with the Army to go through manual COA comparison using the SDT Comparison Matrix. Despite having worked yesterday, apparently they got the matrix into a state in which entering a value caused an exception that prevented population of the totals and prevented selecting another cell unless the matrix window was dismissed and restarted. They ended up exporting the partial inputs to Excel and filling the rest out there. COA1 also won over COA2, scoring 51 to 37, mainly because it was deemed as lower risk. Note that while they had successfully tailored the CCM criteria, they had problems saving the actual CCM values along with the plan.

This was the end of the tool usage. The next hour was spent brainstorming about strategy requirements for the future AOC based on the JEFX process. The main message was that they

want one system that does everything, not multiple overlapping tools each with some specialized features. They said they would refine and prioritize the wish list by the late August meeting with the AFC2ISRC Strategy Requirements Working Group.

Throughout the morning Accenture worked with BAE Systems to debug the output of the translators for the final plan. After much trial and error, they got it working just before acting CFACC Gen. Carlson came in for the End Ex strategy briefing. Unfortunately, Capt. Colazzo did not have time to create the strategy briefing and JAOP in the Accenture composer.

The briefing to Gen Carlson and senior mentor Gen Hurd went fairly well. Gen Croker was very positive about the tools and their ability to give him insights into planning tradeoffs and effects on the enemy. Maj. Greathouse discussed the extensive COG/TS models he had built, the wargaming activity in COG Articulator, and the OAT results. Capt. Colazzo demonstrated the final SDT COA, answered questions about the meaning of each icon such as causal linkages, and pointed out the shortcomings encountered trying to merge offline work back into collaboration. Gen Croker countered that the collaboration capabilities of SDT were very impressive and through a combination of SDT and IWS and frequent teleconferences they were able to coordinate in parallel with the Army much better than they had ever expected.   Gen. Carlson asked why the IPB input link wasn't available when it counted, and Lt. Green defended SDT by explaining that A2IPB upgraded their version without notifying SDT, and that the bug was fixed rapidly once the problem was diagnosed. Ret Maj Wempe complained about the quality of the IPB data, which should have been directed at the IPB tools, but was misinterpreted as a critique of SDT. He apologized afterward about the misunderstanding. Lt. Green afterward suggested that SDT should automatically turn the raw IPB data into a COG Articulator model. Nick acknowledged this would be useful, but would require addressing a difficult natural language understanding problem. However, it might be feasible to automatically create at least the nodes if not the links, given sufficiently structured IPB data as input.

In the afternoon, BAE Systems helped prepare a video production featuring Maj Greathouse, Lt Col Hunerwadel, and Cmdr. Thompson describing the EBO process and lessons learned at JEFX. This would be edited down to a five-minute video to be shown at the C4ISR Summit in October, as part of the JEFX lessons learned discussion. Nick had outlined a script with Cmdr Thompson introducing EBO and the JAEP process, Lt Col Hunerwadel describing Mission Analysis and COA development and Maj Greathouse covering COG modeling and COA analysis. Originally, we had expected that we might get footage of them using the tools, so offline Nick and Brad tried to get their two laptops networked together for collaboration. In the end, after getting a verbal testimonial from each person, they decided that was good enough.

4.3.5   EBO Initiative Assessment Results

All JEFX '04 initiatives are evaluated along two dimensions: technical assessment and operational assessment. The former simply reports how many planned system capabilities were actually demonstrated by the end of the experiments. The latter is more subjective, based mainly on feedback by the players that used the particular initiative tools, but also based on observations made by designated assessors, typically retired AOC officers now working in the AFC2ISRC Air Force Experimentation Office (AFEO). The tech assessment results published by AFEO are given in full in Appendices D. Ops Assessment results are FOUO, but we give a partial summary below.

Because the PBA/EBO initiative contained by far more tools than any other initiative, the EBO portion and PBA portion were scored independently for tech assessment. SDT and OAT performed well on the tech assessment, receiving a Satisfactory rating on almost all pre-planned capabilities. One "Partially Satisfactory" result was in the ability to perform red COG analysis. The explanation refers to Maj. Greathouse's desire to be able to model an exponential degradation in persistence *within* the persistence interval instead of after its expiration. This arguably could be considered a separate capability that was not stated prior to JEFX '04. A second Partial Satisfactory capability was the ability of OAT to receive evidence from AF2TK. The partial grade was due to the need to enter evidence manually and the somewhat counter-intuitive results produced by the OAT evidence algorithm while Lt. Friesen was working with it. Finally, one capability, the ability to export target lists to JTT was graded "Not Demonstrated," but a mitigating note stated that this was not a capability required for operational use by 8th AF (SDT was not used for targeting to begin with).

The ops assessment results were more mixed. Overall, EBO-PBA was given a yellow score. One positive note from Maj. Joos was that the "excellent branch planning effort provided necessary ops utility info to ID required capabilities." A negative result was that "multiple strategy tools hinder process; integration of component and CFC tools will have a huge payoff." This is not necessarily a reflection on SDT, but rather the way JEFX '04 concept of employment called for using at least three tools containing a strategy authoring capability (SDT, IWPC, and SPT). Senior mentor Gen. Croker, who had mainly been supportive of SDT during the experiment, stated that "EBO is a particularly complex concept and needs further refinement. The early CONOPS is quite conceptual and of little practical use at the operational level. Improved working JP 3-30 template a key first step. We have identified user requirements for a future tool set." This implies that many of the problems end-users encountered can be traced to difficulties understanding the *process* for EBO, rather than a problem with imperfect implementation of that process in the tools.

4.4   SDT TRANSITION

In the final contract year following JEFX '04, BAE Systems focused mainly on SDT transition activities, summarized in the following subsections.

4.4.1   Post-JEFX Activities

Since EBO was also one of three main focus areas for JEFX '04, the above results were summarized by Gen. James at the C4ISR Summit in Danvers, MA in October, 2004. Because of our prime involvement in the EBO initiative, BAE Systems was invited to participate in the EBO Panel JEFX '04 Working Group. Our attendance at several working group meetings at Hanscom AFB and Washington, DC helped to inform the group of the positive and negative EBO lessons learned from an insider's perspective, contributing bullets and graphics to Gen. James' presentation. BAE Systems also manned an industry booth at the C4ISR Summit to show an animated presentation of the EBO tools and capabilities developed under the EBO ATD and related programs.

Because of the chief requirement arising from JEFX '04 assessment for a single, unified AOC strategy tool, the initial post-JEFX transition activities consisted of technical interchanges with our counterparts on the IWPC and SPT teams to discuss potential integration and synergies. We attended a TIM at ESC/Hanscom AFB and several teleconferences with Accenture to

estimate tasks and level of effort for incorporating capabilities special to SDT into a future release of SPT in time for TBMCS 1.1.4. We performed a similar integration brainstorming and cost estimation activity with General Dynamics. But with JEFX transition funding reportedly unavailable, neither side wanted to commit to a definite relationship.

Meanwhile, many of the key organizers and players of JEFX '04, including David Hess and Major Dietrich, formed a new Strategy Requirements Working Group (SRWG) to enumerate detailed requirements for the desired unified AOC strategy tool, covering both strategy development and operational assessment. BAE Systems supported many of the SRWG meetings and teleconferences to provide requirements inputs from the perspective of SDT and OAT and our lessons learned from JEFX '04. In recent months, a secondary goal of these meetings was to define a unified EBO data model (plan representation and associated data).

4.4.2   IWPC Integration

Eventually, AFC2ISRC did reach a decision. They recommended that the AOC adopt General Dynamics' IWPC as its general strategy development tool and that specific discriminating capabilities in SDT and SPT be integrated into a future version of IWPC. By early 2005 AFC2ISRC had reached an agreement with AFRL and BAE Systems in which $300K of the OAT program's funding would be applied toward working with General Dynamics on integration of selected SDT and OAT capabilities with IWPC. AFC2ISRC eventually came up with their own supplemental funding for IWPC to both shore up its own IO-related capabilities and to manage the integration of selected capabilities from SDT, SPT, and AFRL-HE's Strategy Visualization contractors. A portion of this supplemental funding was earmarked for SDT, but encountered some contractual delays and was finally received in October 2005. Based on our earlier integration brainstorming, GD decided on integrating SDT's real-time collaboration capability and aspects of its EBO plan representation (prior to this, IWPC's planning screens lacked explicit support for EBO concepts). BAE Systems is now well into implementation of a new "Strategy Collaboration Service" based on the real-time collaboration architecture used in JEFX '04. Some additional enhancements are being developed, such as a "lazy loading" feature that greatly reduces initial download time of plans. We have also worked extensively with General Dynamics and the AFRL-HE Strategy Visualization contractors on a unified EBO data model for the future IWPC release. We have established a good rapport with the IWPC team and prospects for a successful integration in time for JEFX '06 are positive.

4.4.3   JFCOM Transition Opportunities

Until the decision on IWPC from AFC2ISRC, another major transition front pursued by both BAE Systems and AFRL was JFCOM, which was sponsoring a number of programs related to joint-level EBO.

TEBO

BAE Systems received an invitation to attend a Theatre Effects-based Operations (TEBO) User's Conference on April 26-27, in Portsmouth, VA. The audience included the TEBO PM, Technical Lead, and several other government representatives from JFCOM and PACOM. The TEBO lead contractor presented a collection of tools and technologies they had evaluated for potential TEBO inclusion, within the following functional areas: System of Systems Analysis (SOSA), Operational Net Assessment, Effects-based Planning (EBP), Effects-based Execution,

and Effects-based Assessment. SDT was chosen for the EBP category, and had been deployed to a staging lab in South Korea, but had not yet had significant exposure there, possibly because of the lack of TEBO personnel with experience using SDT. TEBO also plans to use SDT for EBP in UFL '05, an annual exercise in South Korea.

To show the applicability of SDT to joint-level planning and the draft TEBO CONOPS, BAE Systems prepared a specialized plan based on an earlier ISPAN demonstration, "Operation Double Jeopardy," in which the U.S. and coalition plan simultaneous global strikes against a North Korean chemical weapons facility and an Al Qaeda training camp in Syria purported to be buyers of the chemicals. The demonstration featured collaborative mission analysis and COA development, and also showed how to author a simple COG model and import a causal chain into the plan (an activity aligned with TEBO's SOSA area). We also briefly discussed lessons learned from our JEFX '04 participation, emphasizing the need to "socialize" EBO methodologies before exposing users to EBO tools. The demonstration and briefing were well-received in general. Based on the draft TEBO CONOPS and discussions at the User Conference, BAE Systems believes that SDT is highly applicable to this CONOPS for SOSA and EBP.

Integrated Battle Command

BAE Systems-AIT has been awarded a prime contractor role on the Integrated Battle Command program, sponsored by DARPA's Advanced Technology Office, aimed at providing advanced adversary modeling and simulation tools for joint-level EBO. JFCOM is providing subject matter experts and organizing end-user experiments at their J9 experimental facility. Our technical approach leverages SDT and the other EBO ATD tools described in this report, as well as a number of other in-house and subcontractor-provided modeling and simulation tools.

MNE4

In early 2005, AFRL expressed strong interest in supporting a major multi-national experiment (MNE) evaluating EBO tools called MNE4. They asked BAE Systems to prepare to attend a tool evaluation meeting for MNE4 at JFCOM-J9 in early March. Unlike JEFX '04, due to unavailability of experiment funding for participating tools, BAE Systems support would have been limited to travel to planning meetings and on-site training and technical support. No new development was planned due to limited resources. However, once AFC2ISRC's decision to integrate portions of SDT with IWPC was announced, AFRL abandoned plans to participate in MNE4 in favor of using their remaining funding toward the IWPC integration.

4.4.4 Joint Coordinated Real-Time Engagement (JCRE)

In late 2004, BAE Systems was contacted by SETA representatives of the ONR JCRE ACTD, which is focused on tactical-to-operational-level agile planning for special operations forces. The JCRE program had been recommended by one of its operational sponsors to look for effects-based planning and execution technology to help STRATCOM, another JCRE target site, in its new missions. On December 3, 2004, we hosted a meeting at our main Burlington, MA office with JCRE SETA representatives. The JCRE SETA representatives contacted BAE Systems again a month later to invite us to attend initial teleconferences to discuss pre-program planning.

After attending these planning meetings BAE Systems-AIT was soon brought onto the core team to assist the prime integrator with system engineering tasks relating to incorporating EBO concepts into the JCRE data model in 2005. Our successful contributions to the kickoff meeting and subsequent data model discussions led by late summer to an expanded role that now includes development and integration.

4.5   LESSONS LEARNED

The SDT team met internally shortly after JEFX '04 to discuss and record lessons learned, both technical and programmatic, spanning from the early research to the fever pace of JEFX '04 technical support.

4.5.1   EBO ATD Lessons

The following are only a representative sample of the lessons learned throughout the first three years of SDT research under the EBO ATD prior to involvement in JEFX '04.

Scenario Storyboarding

The original SDT proposal provided more than enough detail from which to derive software requirements for early prototypes. However, those requirements would have been difficult to prioritize without a concrete scenario to drive them. Fortunately, AFRL had prepared a detailed challenge problem, Operation Deny Force, which served as inspiration for ideas for an end-to-end storyboard covering SDT, CAT, and Endstate. While aspects of the early storyboard may have seemed overly contrived, they helped focus initial development on the components and capabilities most critical for conveying the long-term vision of our efforts: the Plan Editor, COG Analyzer (CAT), and Endstate TSA. A joint storyboard also helped convey how the tools would feed information to each other, even if these inputs were canned or manually entered during early demonstrations. We also encountered a risk of developing to storyboards, namely that the most ambitious features in the storyboard may eventually prove impractical to realize in software.

Model Abstraction

Perhaps the hardest lesson we learned was the difficulty in designing meaningful interoperability between two very different modeling approaches. Our original vision for Endstate was not only to augment the strategic plan with operational-level effects and targets (which it did by the end of the program) but to also augment the strategic COG model with an operational-level causal fragment. This fragment was originally envisioned as an abstraction, or summary, of the portions of the operational network model impacted by chosen targets. While we were able to develop specific examples of this in our storyboard, the nature of Endstate's closed loop network flow optimizations did not easily lend itself to a meaningful Bayesian network representation that preserved the flow semantics of the original network. We would either have to break the original semantics and settle for an abstraction that might often lead to results inconsistent with the original model or map the continuous Endstate output variables to Boolean states via arbitrary thresholds that might mask important effects that happen to fall short of the thresholds.

Unified data model

Another key lesson was the importance of achieving a unified data model for plan representation as early as possible. SDT's initial versions featured a purely declarative plan representation through a combination of named property hashtables and grammar-based natural language templates. This allowed us to rapidly evolve the plan representation as we learned more about the domain from SMEs and doctrinal studies. However, it also hampered incorporation of third-party components from co-contractors such as ISX, who typically had their own specialized data model. This required a significant amount of extra effort to develop and maintain translators between the generic but extensible SDT representation and the more specific but more rigid ISX representation. After SDT abandoned its original generic effect-task pair representation and adopted the core EBO ontology, interoperability of the plan structure was significantly simplified, but interoperability at the individual node level was still cumbersome, though functional. Not until the migration from the natural language editors to the conventional forms-based user interface was a unified Java-based model feasible. Encapsulating the data model using abstract interfaces was the final piece of the puzzle, allowing us to reuse and extend the ISX concrete implementation in a way that enabled seamless viewing and editing of the same data objects in both ISX-provided editors and native SDT editors.

Non-kinetic Actions and Effects

Finally, one important lesson learned in hindsight is that our demonstration scenarios, while appropriate for showcasing the capabilities of the combined tools (SDT, OAT, and Endstate), may have focused too heavily on kinetic actions and effects achieved through judicious use of airpower. While the final version of SDT does include mission and strategy templates covering non-kinetic mechanisms such as deception or information operations, we did not elevate these plan fragments front-and-center during demonstrations, leaving some with the misperception that SDT is limited to or biased toward kinetic aspects of EBO. The COG Articulator leadership model highlights the potential for non-kinetic effects by including nodes relating the population and support of political allies to the enemy's willingness to deploy WMD. Adding a "deterrence" phase to the Operation Deny Force SDT demonstration plan that exploits these non-kinetic nodes might have helped dispel these false perceptions of bias toward kinetic mechanisms.

4.5.2 JEFX '04 Lessons

Training

As part of each JEFX Spiral and Main Ex., users and assessors received training on EBO EndState capabilities as part of the SDT training activities. Lessons learned from the JEFX training sessions point to the need for more individualized training and a more favorable training environment. SDT training for JEFX '04 was conducted in a noisy environment with trainees attempting to follow training directions via a headset communication system. A typical training class consisted of 10-15 trainees under the direction of a single tutor. While BAE Systems provided additional training facilitators to answer questions, it was difficult for participants to catch-up with the training after suffering a fall-back. In addition to high trainee/tutor ratios, there was an issue of participant interest. While some trainees required instruction on use of the entire system, most users were focused on a specific task or function of the software. This made it difficult to hold the interest of the trainees. Future training sessions would benefit from more

individualized training geared toward smaller groups and focused on activities that would be conducted by the group.

Effects-based Planning

The SDT Plan Editor was the tool most widely used by the Strategy Division players at JEFX '04. Hence, AFRL's decision to allow BAE Systems to invest heavily in improving its ease of use and meeting or exceeding coverage of comparable data items in the competing infrastructure strategy tools paid large dividends. While not without minor bugs, SDT's combination of a hierarchical Tree View with a graph-based Structural Editor proved more appealing to users than SPT, which lacked an explicit hierarchical view (plan elements in its Timeline View could only be indented manually), and IWPC, which lacked a graph-based view (aside from its COAST tool, which shows a more complex blue-on-red schematic view similar to that of COG Articulator). The Tree View is critical for navigating large plans that are too large to fit in a single graph-based view, allowing users to collapse irrelevant portions of the tree and expand regions of interest while retaining the surrounding high-level nodes for visual context. The graph-based view is better suited for viewing small plans or drilling down within a large plan to a single fragment at a time. Combining these two views seamlessly provides users the best of both worlds.

One negative lesson learned in hindsight that we would apply if invited to future experiments of size comparable to JEFX, is to be more cautious in responding to user-requested bug fixes and enhancements. Especially after Spiral 2, the first major trial at Nellis AFB, BAE Systems worked extra hard to tick off as many bugs and features out of a list at least fifty items long. Many of the feature requests were contradictory, and some were probably a matter of individual preference, with questionable wider appeal. We tried to weed these out, but the temptation to "overfit" the tool to the small sample of 8th AF JEFX '04 players was strong. Of course, these efforts had a short-term positive effect that the users were pleased, indeed often amazed at the improvements delivered each subsequent spiral. However, a negative side effect was increased technical risk, and instances of a "ripple effect" of certain changes leading to or uncovering other bugs in other areas of the system.

Adversary Modeling

The most apt lesson for the adversary modeling tool, COG Articulator, was that *not all users are created equal.* From the very beginning the Strategy Chief was a skeptic, stating that these kinds of models cannot be properly validated but if it is done at all, adversary modeling should remain strictly in the ISR Division's jurisdiction. In contrast, we could not have asked for a better advocate than Major Greathouse, the Deputy Ops Assessment Chief. While obviously a "power-user," who professed to having a limited amount of prior modeling experience, Maj. Greathouse built complex system of systems models that far exceeded the original expectations of its designers. In fact, he came up with novel applications of COG Articulator for building new types of models other than the dynamic causal "influence" networks for which it was intended. For example, he used it to build if-then style decision trees, blue-on-red wargaming constructs, and top-down COG decompositions based on the Strange methodology (Strange, 1996) (see Figure 48). More details on these modeling techniques are provided in (Pioch and Greathouse, 2005).
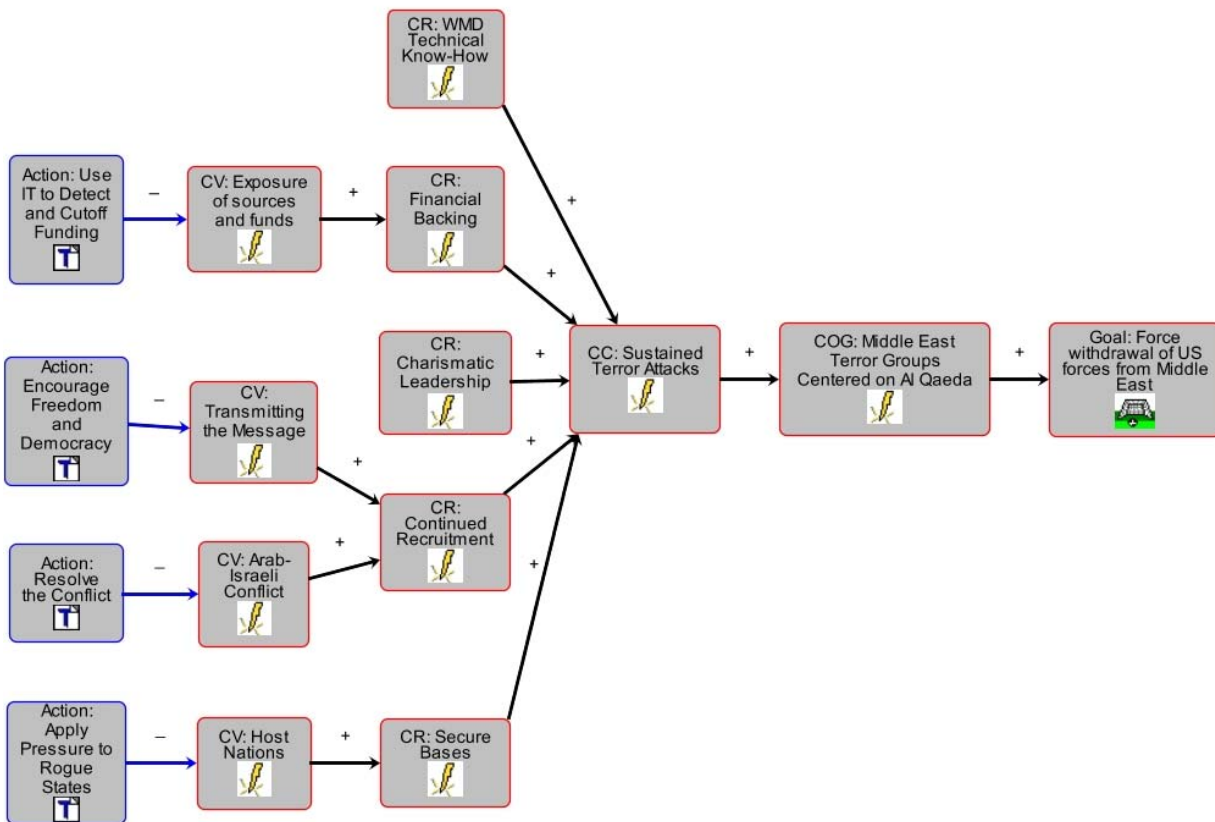
Figure 48. Example of a Strange method COG model in COG Articulator.

By the end of Main Ex, Maj. Greathouse's infectious enthusiasm for COG modeling had begun to win converts among other 8th AF players originally reluctant to build explicit models because of the extra workload. These users were experiencing the benefits of adversary modeling:

- It helped the strategy operators arrive at a consensus on assumptions about the relevant enemy COGs and their inter-dependencies and vulnerabilities.

- It serves as an explanation of the blue strategy, grounded in terms of its impact on red COGs and supporting systems.

- It helps planners avoid undesired effects to the extent that the models account for them.

Despite these achievements, COG Articulator exhibited some problems interacting with OAT during JEFX that could have been discovered earlier if BAE Systems had performed scalability tests with sufficiently large plans and models. Use of globally unique identifiers for all COG nodes should have been adopted early on, preventing node ID conflicts that occurred when importing COG model fragments. Most importantly, if we could start JEFX over with more time to prepare, we would have limited OAT's role to be the underlying inference engine underneath COG Articulator, rather than imposing two separate user interfaces, one for modeling and one for analysis. Much time and effort was wasted by operators moving back and forth between these tools and re-entering model data lost after a crash. This mirrored the broader findings of

the JEFX assessors on the inherent problems with imposing multiple strategy tools for similar functions. A unified single user interface for modeling and analysis could have been achieved simply by developing a probability profiles view in COG Articulator and hooking it up to probability queries to the OAT engine.

Collaboration

The expected advantages of SDT's original delegation-based collaboration approach were three-fold. First, it allowed users to explicitly manage who is working on what portions of the plan. Second, it provided traceability of a plan's evolution through logging of messages and workflow transitions. Finally, implementation was relatively straightforward, building on an existing SDT capability to save and import plan fragments as XML files.

In practice, when this capability was demonstrated during Spiral 2, the first major on-site round of JEFX experimentation, users pointed out significant disadvantages that hindered their exploration and adoption of the collaboration tools. Already feeling overwhelmed by the need to rapidly train to use multiple strategy authoring tools at JEFX, *users objected to the additional workload imposed by the workflow process of managing delegated plan fragments*. While SDT provided tools to help track these delegated fragments, *users did not like the idea of having proliferated copies of the same plan embedded in the task boxes of multiple users with no single common view of the entire plan* until all fragments had been approved and incorporated. Users also disliked having yet another login on top of the accounts imposed by the operating system and other JEFX tools. Finally, by necessity, the user typically playing the role of plan integrator tends to be someone of high authority familiar with all aspects of the strategy, such as the strategy chief. These collaboration tools would impose an additional burden on the already busy chief, having the unintended side effects of *making the chief a bottleneck and delaying the rest of the team from seeing each other's merged plan contributions*.

To help overcome these problems, the strategy team gave the following guidelines for useful strategy tool collaboration features:

- Usernames should be inferred from the operating system user account login

- Collaboration should be transparent, using the same authoring methods as with stand-alone planning.

- Multiple users should be able to contribute simultaneously to the same plan.

- Every user's view of the plan should be updated in real-time with pointers to what has changed.

- There should be one single, common version of the plan at any given time.

Based on these guidelines, we spent over three months and significant manpower between Spiral 2 and Main Ex working to realize these guidelines in a new real-time collaborative planning capability.

This improved collaborative planning capability was put to the test during the final JEFX '04 Main Ex spiral. Because of its experimental prototype status, its use was limited to a four-day branch planning activity involving a smaller strategy group working on a notional contingency plan that would not affect any downstream JEFX teams or processes. This branch planning

activity involved distributed coordination between the Joint Force Commander (JFC) at Nellis AFB, the Air Component strategy plans team also housed at Nellis, and a smaller Ground Component strategy team at Hurlburt AFB.

Prior to branch planning, the person playing the role of JFC manually authored a small high-level plan skeleton intended as a common baseline for the Air and Ground Components. Because of uncertainty in SDT's ability to support collaboration over a wide-area network, we set up each site with its own collaboration server and database so that the two components could work independently of each other. However, we also configured one of the Hurlburt clients to be able to download and upload plans from the Nellis server so that the two components could loosely monitor each other's work. In addition, third-party collaborative tools such as Information Workspace (IWS) were available to enable shared displays, virtual conference rooms, audio broadcast channels, and chat. The third-party shared display capability turned out to be useful in cross-site collaboration because it provided a common visual reference for discussion during end-of-day hotwash teleconferences. *The combination of these capabilities led the JFC to declare that this year's group achieved a higher degree of cross-service strategy collaboration than seen at any previous JEFX.*

The branch planning teams followed the same multi-step JAEP used in the previous spirals, starting with Mission Analysis, proceeding with COA Development, Analysis, Wargaming, and Selection. The tab-by-tab collaboration capability worked very well during Mission Analysis. During previous spirals, each user had to work in SDT on a separate local plan file, so all contributions to the Mission Analysis tabs had to be sent via chat tool and manually pasted by the Strategy Chief into the master plan. Now, each user could work in parallel on filling in the statements for each separate tab with updates going into the master plan automatically via the database. This *greatly reduced the time and effort needed to complete the Mission Analysis phase, leaving users with a positive impression of the capability.*

As users proceeded with the next step, collaborative COA development, results were more mixed. Here, the Nellis Air Component broke into two teams, each working on alternate COAs, with plan updates going directly into the common database and pushed out to all users' displays. This worked well at first while the plan was still relatively small. However, as the plan grew to over a hundred nodes, the download wait time increased to over ten minutes, and was often exacerbated by high network load. *When users' patience was taxed by the high download time, they tended to resort to working in off-line mode.* This in turn required them to periodically synchronize manually by uploading each team's plan, copying one COA to the clipboard, and pasting it into the other plan. If one team worked off-line while the other worked collaboratively, the off-line changes would incur yet another wait interval to download and merge the changes. Thus, *the download time introduced a different type of process bottleneck* than seen with message-based collaboration. Eventually, the team abandoned real-time collaboration and developed the remainder of the COAs as separate off-line plan files. This experience led to the decision to implement a "lazy loading" feature for the Strategy Collaboration Service currently being developed by BAE Systems for IWPC, in which the tool only downloads detailed plan element information when the user moves to a view that calls for it.

The SDT development team performed significant functional and stress testing prior to Main Ex, which led to detection and correction of several major bugs and documented workarounds for other bugs that could not be fixed in time for deployment. *Despite having trained the users*

on these known limitations, there were several instances in which users tried them anyway, leading to occasional lost work. For example, copy or cut and paste operations on a certain node type would delete nearby plan fragments. Also, copy and paste operations sometimes used shared objects rather than separate copies of the pasted fragment, such that edits to the pasted fragment interfered with earlier work. In hindsight, taking the time to disable known buggy actions during collaborative mode would have helped prevent these errors. *Training alone was insufficient motivation to remove the temptation to use a feature that is active in the toolbar.*

Overall, the strategy team's feedback was positive regarding the intended style of collaboration provided for Main Ex. It was significantly better received than the original message-based approach, with *highest praise going to the Mission Analysis tab-by-tab locking.* The *slow download time was noted as the primary weakness, followed by the lack of automated merge capability for off-line work*. During the final hotwash, the strategy team listed real-time collaboration as a minimum requirement for future AOC strategy tool procurements, indicating that the design fit well with their operational needs, even if the implementation fell a bit short. Even so, they observed that *the embedded strategy tool collaboration works best when complemented by the generic collaborative tools for display sharing and chat.* Moreover, *collaborative tools are no substitution for face-to-face coordination*, a lesson learned in Spiral 3, when the team became productive only after moving out of the cacophonous central AOC environment where they were forced to collaborate only through tools to a separate war room where they could discuss ideas face-to-face.

General Lessons

One of the most important general lessons from JEFX '04 was to meet and develop a rapport with the end-users as early as possible. Learn their operational process and team roles and responsibilities, then work to tailor the tools to align with their process and roles. Expose the users to the tools as early as possible through demonstrations and briefings. Our initial visit to 8th AF at Barksdale was over four months before Spiral 2, and was followed up with two additional events for training and familiarization. This early interaction also gave us time to incorporate features addressing highest priority aspects of their feedback by the start of Spiral 2.

Second, it is impossible to underestimate the importance of unit- and system-level testing. When faced with the short timelines and high pressures of JEFX, some projects might be tempted to throw off the burden of a managed software development process in favor of agile methods of programming. In our experience, the process stages of requirements analysis, design, and especially testing became even more critical in that they helped us focus our limited manpower on the highest priority tasks and caught many bugs early enough to address them before deployment. Before each spiral, our goal was to leave at least a two-week window for system testing, with a code freeze beforehand for everything other than approved bug fixes. During setup week for each spiral, getting at least two team members on-site as early as possible to test in the real environment helped identify environment-specific bugs and lay down a stable foundation for experiment execution. The main area in which our testing had significant room for improvement was scalability testing, especially regarding the export of plans and models among SDT, COG Articulator, and OAT, and the timeliness of initial database plan download for collaboration.

While many users came to appreciate the EBO tools' ability to provide insights and shared understanding of the enemy, the majority of users were uncomfortable with the added

complexity not only of these tools but of the whole EBO process compared to the simpler planning methods they had been taught prior to JEFX.   More generally, *the EBO methodology needs to be better defined in doctrine and entrenched in the AOC strategy schools* before it is likely to be widely accepted in a large-scale experiment venue such as JEFX. Many users had false expectations that EBO should make planning faster and easier, when in fact it is more labor-intensive for both intelligence preparation and planning, but has the potential to lead to better products in both areas with more timely and efficient use of resources to achieve strategic effects.

SECTION 5

CONCLUDING REMARKS

During this project, we developed and integrated two major EBO capabilities within SDT: a suite of effects-based plan authoring tools collectively known as the Plan Editor, and a suite of adversary modeling and analysis tools. We refined these tools through a series of demonstrations and experiments beginning with EBO ATD PMRs and culminating in JEFX '04. In the final year we worked to transition these capabilities to both operational use and follow-on research programs.

5.1   EBO PLAN AUTHORING

The centerpiece of our SDT research was the Plan Editor, a collection of integrated effects-based plan authoring tools. We began by prototyping a hierarchical effect-task editor combining a Tree View for hierarchical plan display and a restricted natural language statement editor for detailed description of effects, tasks, and mechanisms in the plan. We continually refined the core ontology for the Tree View and Properties Editor, initially aligning with the ISX plan ontology, and later extending the representation to support new plan element types such as undesired effects and new attributes needed for effects-based JAEP in JEFX '04. We also integrated alternative views using a model-view-controller design pattern, starting with ISX's timeline, objective, and phase editors, which later were replaced with our own Structural and Timeline editors tailored for strategy operators at JEFX '04. With the help of Dr. McCrabb, we built an effects dictionary reference to convey the semantics of library of effect verbs, along with sample statements showing how they are applied. Using doctrinal sources, we authored a core set of mission templates for common Air Force functions, which were later refined for JEFX '04 by Blue Force subject matter experts. We integrated ISX's strategy template authoring capability and a corresponding suite of built-in strategy templates. Finally, BAE Systems developed prototype editor screens for planned and actual assessment, including both completion level and schedule status of plan elements, with special indicators added to the Timeline view for intuitive visualization of schedule status.

To prepare for JEFX '04, we explored and prototyped two independent methods of collaborative planning. The first method used a delegation-based workflow in which authors could assign plan fragments to designated specialists for elaboration. An new set of email-like views were developed for sending messages with embedded plan fragments for both tasking a user and submitting fragments back to users. Based on feedback by the strategy operators, we eventually replaced these tools with a more transparent, real-time collaboration capability in which a central database plan representation is maintained and multiple users may edit different portions of the same plan simultaneously. Locking, change notifications, and updating of plan views are handled automatically by the system.

5.2   ADVERSARY MODELING AND ANALYSIS

During the first two years of research, we explored two alternative uses for the AFRL's CAT. The first application was direct prediction of the timing and probability of blue COA success. To achieve this, we added qualitative fields in certain SDT plan element editors relating to prior

task success probability, causal strength of mechanisms, and delay and persistence durations. Then we developed an algorithm to translate the COA into a CAT dynamic Bayesian network. CAT could then be used to predict and display probability profiles for any plan node, including the top-level COA.

At the same time, we explored the use of CAT as a COG model authoring tool. We built simple examples of COG models for POL and Leadership using red-outlined nodes and causal links to represent cross-COG and intra-COG dependencies. We showed how CAT could be used to compare the baseline probability of enemy goal achievement to that of various alternative blue COAs, represented by combinations of blue intervention nodes causally inhibiting certain red nodes at scheduled times.

Based on our findings in CAT, we developed a stand-alone COG Articulator tool specifically for the purpose of COG model authoring. This provided more explicit support for COG node authoring, allowing users to choose from a number of COG taxonomies such as Warden, Barlow, or PMESII [refs]. It used a family of icons to distinguish between enemy goals, conditions, resources, actions, and beliefs. It retained the core probability concepts used in CAT, but with terminology appropriate to AOC strategy or intelligence operators. We tightly integrated COG Articulator with CAT by extending the existing SDT-CAT interface layer developed during the first year. Users would be able to author detailed models in COG Articulator, and freely move to CAT for probability analysis, then to COG Articulator to refine the model.

BAE Systems also integrated SDT with the two operational-level analysis capabilities of SDT's sister program, Endstate. We developed an initial loosely-coupled XML message exchange in which SDT sends a summary of the plan and its direct and affected targets and Endstate runs a workarounds analysis on the relevant target systems to predict outages and reconstitution times for affected nodes of interest. Later, this capability was integrated more tightly and the demonstration became significantly more cohesive with the development of a map-based Target Set Tool and Query Tool. This allowed the user to interactively find appropriate direct and affected targets and append them to tasks and effects in the SDT plan. It also allowed SDT to highlight nodes in the Tree View for which Endstate outage profiles were reported, and display the outage information in a chart on demand. Finally, after Endstate developed an option generation capability for automatic recommendation of strike targets, we developed new XML messages appropriate for option generation and new user interface mechanisms for incorporating the resulting user-approved targets.

5.3   DEMONSTRATIONS AND EXPERIMENTS

During the first three years of research, we periodically demonstrated the above evolving capabilities at EBO ATD PMRs and special venues such as the AF SAB conference at Langley AFB. These activities presented valuable opportunities to work directly with EBO co-contractors such as ISX and subject matter experts from other government research sites and the operational community. The feedback from these demonstrations helped us refine the core capabilities. One activity in particular, the AFC2ISRC-sponsored EBO Workout, held at the Langley AFB CAOC-X Transformation Center in April, 2003, provided a unique knowledge acquisition opportunity. SDT representatives watched a team of senior retired AOC operators walk through an effects-based version of the AOC planning process. The SDT team worked in parallel to "shadow" their activities in the EBO ATD tools, helping to both validate many existing capabilities and reveal new areas requiring further exploration.

Soon after the EBO Workout, BAE Systems assisted AFRL with their JEFX '04 PBA/EBO initiative proposal, which was accepted in mid 2003. This touched off a new phase in the project in which the existing plan authoring and adversary modeling tools would be refined and hardened for deployment at three major JEFX spirals in spring and summer 2004. Significant new capabilities would be added, including collaborative planning and a bi-directional interface with TBMCS Strategy Management Services for interoperability with other JEFX strategy tools. Early interactions with the 8th AF strategy division operators provided invaluable early feedback, much of which was incorporated into the initial delivery for Spiral 2, the first major experiment spiral. Based on user feedback, we migrated from a delegation-based collaboration approach to a transparent, real-time collaboration architecture in which multiple users may work in parallel on the same plan with instantaneous change updates. In addition to this new development, the SDT team provided extensive on-site support to install the tools at Nellis and Hurlburt AFBs, provide tool training, and help users with questions and problems arising in the EBO tools.

Overall, end-user feedback on SDT by the end of JEFX '04 was positive, and it was seen as an improvement over currently available AOC strategy tools. However, the reaction to the EBO-specific capabilities was mixed. Many users were uncomfortable with still-evolving EBO planning process, seeing it as adding workload beyond the strategy-to-task methods on which they had been trained. These users tended to be reluctant to use the new plan element concepts in SDT like *effect* and *causal linkage,* and steered clear of adversary modeling and analysis. In contrast, other users were more receptive to these new tools, with one user in particular building and analyzing extensive COG models that far exceeded our original expectations.

5.4   SDT TRANSITION

Despite the individual positive testimonials of EBO devotees at JEFX '04, the overall assessment of the utility of both the EBO focus area and the tools initiative was mixed. SDT's non-EBO capabilities overlapped with two other competing strategy tools, IWPC and SPT; a somewhat artificial concept of operations that forced users to work with all three of these tools at different stages exacerbated the frustration with a process already more intensive than the strategy-to-task process to which they were accustomed. It was clear after JEFX '04 that AFC2ISRC wanted a single overarching strategy tool for the future AOC weapons system. In early 2005 the Center designated IWPC, the oldest and most operationally mature of the tools, already deployed in the ISR Division, to be adopted by the AOC Strategy Division for general strategy planning. Additional funding from the Center supplemented devotion of part of AFRL's OAT funding to bring BAE Systems onto the General Dynamics team to assist with a unified EBO plan data model and develop a collaborative planning service based on SDT's real-time collaboration. This is expected to be the primary path for transitioning SDT technology to AOC operational use.

Several other follow-on research opportunities have been pursued, with some early success. BAE Systems is integrating SDT and OAT into the ONR-sponsored JCRE ACTD and is helping to enhance existing tools for SOF planning with effects-based concepts. We have won a Phase I prime contractor role on DARPA's Integrated Battle Command, which focuses on multi-resolution modeling and simulation tools for JFCOM intelligence cells embedded with combatant commands. BAE Systems is actively pursuing other research opportunities at JFCOM, AFRL, and other joint and service-specific laboratories.

SECTION 6

RECOMMENDATIONS

Based on the outcomes of this project, we recommend further research and development in effects-based planning, modeling, and analysis.

System of Systems Analysis: SDT's technical approach to COG modeling turned out to be a forerunner of a broader activity dubbed by JFCOM as System of Systems Analysis (SOSA). This involves detailed studies of the "nodes" comprising individual enemy systems, and the linkages (causal or otherwise) among them in terms of both physical structure, operational process, and other dependencies. JFCOM has developed an extensive concept of operations for SOSA, but to date few effective tools have been developed to facilitate the key processes of authoring SOSA models and later querying and analyzing them to determine which actions and resources to apply against which enemy nodes to achieve desired effects. While IBC is one program actively exploring this area, more research is needed on *qualitative* modeling methods to reduce the authoring burden on intelligence operators. For example, users uncomfortable with assigning specific probability estimates in COG Articulator might benefit from more qualitative Bayesian abstraction frameworks such as Pearl's $\in$-calculus (Goldszmidt and Pearl, 1996) or Lucas' Qualitative Probabilistic Networks (Lucas, 1996), both of which eliminate the need to specify numerical probabilities.

Automated Planning: SDT's ability to import causal chains from COG Articulator as blue plan fragments only scratches the surface of rich possibilities for automated planning algorithms, given sufficiently detailed adversary models. More research is needed to formulate and explore solutions to an optimization problem to determine appropriate combinations of blue interventions against low-level enemy nodes in order to achieve a specified set of goal desired effects on high-level red nodes. Initial solutions could employ heuristic algorithms that determine the reachable paths in the enemy model between possible interventions and nodes representing objects of effect, recommending those paths with the highest cumulative causal strength, lowest cumulative delay, and highest cumulative persistence. More sophisticated algorithms might employ iterative hill-climbing techniques such as Maximum Marginal Return-Closed Loop (Shaw et al, 1994) or genetic algorithms (Franklin, 2002) to explore the vast search space of intervention combinations and timing.

Model Abstraction: During the early years, we explored potential model abstraction techniques for augmenting strategic COG models encoded as causal DBNs with operational-level fragments encapsulating the relevant portions of the detailed target system network flow representation. In the end we determined that further research would be needed to mitigate the potential negative side effects of model divergence or masking of effects that could result from these techniques. One possibility would be to adopt a rule-based system rather than Bayesian networks as the overarching abstraction framework, with each detailed model compiling a summary of relevant dynamic system behavior in the form of if-then rules. This leads to another key research issue in model abstraction: what is the appropriate ontology for capturing semantics of the model inputs and outputs to enable seamless interchange between widely varying model representations? Again, IBC is one program in which BAE Systems will be investigating operationally-focused solutions to this problem, but we believe additional basic research in model abstraction would greatly benefit the Air Force and other services.

SECTION 7

REFERENCES

Franklin, D. Data Miners, *Time (Online Edition),* December 23, 2002, http://www.knowledgecc.com/news1.htm.

Goldszmidt, M. and Pearl, J. "Qualitative Probabilities for Default Reasoning, Belief Revision, and Causal Modeling", *Artificial Intelligence*, Vol. 84, No. 1-2, pp 57-112, 1996.

Joint Publication 3-30. *Command and Control for Joint Air Operations*, Chapter 3, pp. III-4-III-17, June 5, 2003.

Jones, E. K. "Continuous Strategy Development for EBO," Technical Proposal, P-3800-136, BAE Systems (formerly ALPHATECH, Inc.), September, 2000.

Lucas, P. J. F. "Bayesian network modelling through qualitative patterns," *Artificial Intelligence*, Vol. 84, No. 1-2, pp 57-112, 1996.

Pioch, N. "SDT Software Design Document," BAE Systems (formerly ALPHATECH, Inc.), v. 1.3, November, 2004.

Pioch, N. J., Daniels, T., and Pielech, B. "Collaborative Effects-based Planning using Adversary Models and Target Set Optimization," *SPIE Defense and Security – Enabling Technologies for Simulation Science VIII*, 2004.

Pioch, N. J. and Greathouse, S. "Collaborative Effects-Based Planning and Adversary Modeling: Lessons Learned from JEFX '04," *SPIE Defense and Security – Enabling Technologies for Simulation Science IX*, 2005.

Shaw, J.J. Gerber, M.A., James, R.M., and Rizzuto, A.P. The Look-Ahead Battle Planner. Proceedings of the ADPA Conference on BMC3, March 23-25, 1994, pp 141-146.

Strange, J. "Centers of Gavity and Critical Vulnerabilities: Building on the Clausewitzian Foundation So That We Can All Speak the Same Language." *Perspectives on Warfighting,* Number 4, Marine Corps University, 1996.

Technical Report 1071 Active Templates Final Report, BAE Systems (formerly ALPHATECH, Inc.), July, 2004.

"Universal Joint Task List (UJTL)," Joint Chiefs of Staff, Washington, D.C., July, 2002.

Warden III, John A. *The Air Campaign: Planning for Combat.* Pergamon-Brassey's, 1998.

White, C. M., Prendergast, M., Pioch, N., Jones, E. K., and Graham, S. "Effects-based Strategy Development Through Center of Gravity Analysis and Target System Analysis," *SPIE Aerosense*, 2003.

APPENDIX A

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

Acronyms not spelled out in the main text of this report are defined here.

A2IPB	Automated Assistance for IPB
AAA	Anti-Aircraft Artillery
AD	Air Defense
AFC2ISRC	Air Force Command and Control Information Surveillance and Reconnaissance Center
AFRL	Air Force Research Lab
AF SAB	Air Force Scientific Advisory Board
AFDD	Air Force Doctrine Document
AOC	Air Operations Center
API	Application Programmer's Interface
APS	Adaptive Planning Software
ATD	Advanced Technology Demonstration
BE	Basic Encyclopedia
C2	Command and Control
CAT	Causal Analysis Tool
CCIR	Commander's Critical Information Requirements
CCM	COA Comparison Matrix
CFACC	Combined Forces Air Component Commander
CFC	Combined Forces Commander
CFLCC	Combined Forces Land Component Commander
CI	Commander's Intent
COA	Course of Action
COG	Center of Gravity
CSAR	Combat Search and Rescue
CSC	Computer Software Component
CSU	Computer Software Unit
CTL	Candidate Target List

DAML	DARPA Agent Markup Language
DARPA	Defense Advanced Research Projects Agency
DBN	Dynamic Bayesian Network
DCG	Definite Clause Grammar
DCOAD	Dynamic Course of Action Decision Tool
DIME	Diplomatic-Information-Military-Economic
DNA	Do Not Affect
DNS	Do Not Strike
DODAF	Department of Defense Architecture Framework
EB-JAEP	Effects-based Joint Air Estimate Process
EBO	Effects-based Operations
EBOWS	EBO Wargaming System
EP	Electric Power
ESC	Electronic Systems Center
GIS	Geographic Information System
GUI	Graphical User Interface
IADS	Integrated Air Defense System
IO	Information Operations
IPB	Intelligence Preparation of the Battlefield
IWPC	Information Warfare Planning Capability
IWS	Information Workspace
J2EE	Java 2 Enterprise Edition
JAEP	Joint Air Estimate Process
JAG	Judge Advocate General
JATO	Joint Air Tasking Order (process)
JCRE	Joint Real-time Coordinated Engagement
JDP	Joint Defensive Planner
JEFX	Joint Expeditionary Forces Experiment
JFACC	Joint Force Air Component Commander
JFC	Joint Force Commander

JFCOM	Joint Forces Command
JMEM	Joint Munition Effectiveness Manual
JMS	Java Messaging System
JP	Joint Pub
JTT	Joint Targeting Toolbox
JWAC	Joint Warfare Analysis Center
LOCS	Lines of Communication
MA	Mission Analysis
MDMP	Military Decision Making Process
MIDB	Modernized Integrated Database
MOE	Measure of Effectiveness
MOP	Measure of Performance
MSEL	Major Scenario Event List
NAI	Named Area of Interest
NL	Natural Language
OAT	Operational Assessment Tool
OCA	Offensive Counter-Air
OV	Operational View
PBA	Predictive Battlespace Awareness
PMR	Program Management Review
POL	Petroleum-Oil-Lubricants
ROE	Rules of Engagement
SAM	Surface-to-Air Missile
SDT	Strategy Development Tool
SEAD	Suppression of Enemy Air Defenses
SME	Subject Matter Expert
SMS	Strategy Management System
SOAP	Simple Object Access Protocol
SOF	Special Operations Forces
SOSA	System of Systems Analysis

SPT   Strategy Planning Tool

SRWG   Strategy Requirements Working Group

STO   Software Task Order

STT   Strategy-to-Task

SV   System View

TAI   Target Area of Interest

TBM   Theatre Ballistic Missile

TBMCS   Theatre Battle Management Core System

TEBO   Theatre Effects-based Operations

TSA   Target Systems Analysis

UI   User Interface

UID   Unique Identifier

UJITL   Unified Joint Integrated Task List

UML   Unified Modeling Language

WARNORD   Warning Order

WMD   Weapons of Mass Destruction

WOE   Weight of Effort

XML   eXtensible Markup Language

XSLT   eXtensible Stylesheet Language Translation

APPENDIX B

SDT CROSS THRUST DEMONSTRATION SCRIPT

SDT Demonstration Script

Prepared by:
Nick Pioch (npioch@alphatech.com),
Troy Daniels (tdaniels@alphatech.com)
For EBO SDT 1.4.1 (Oct. 27, 2003)
Document Version: 1.4 (Oct. 27, 2003)

1. Introduction

This document walks through the process of demonstrating the Strategy Development Tool (SDT) and Causal Analysis Tool (CAT) for AFRL's Effects-Based Operations Advanced Technology Demonstration. It contains both instructions for navigating the SDT and CAT user interfaces, along with a narration describing the primary planning and assessment capabilities. Navigation instructions are in italics, interspersed throughout the narration.

The document includes both a 15-minute and 30-minute version of the demonstration. The 15-minute demonstration covers only the SDT and CAT. The 30-minute version includes expanded features of SDT and Target Systems Analysis. Sections pertaining only to the 30-minute demonstration are underlined.

2. Setup

Installation:
1. If the SDT auto-installer is used, double-click on the installer and follow its instructions. Otherwise, unzip the runtime distribution and follow SDT installation instructions in INSTALL.txt.
2. To install the right version of CAT for SDT, go to the SDT's bin directory, and double-click on the CATInstaller_2003_0123.exe, then follow the installer instructions.

Running:
3. If SDT was installed via the installer, run SDT via Start Menu->Programs->Strategy Development Tool. Otherwise, go to %SDTHOME%\bin and double-click runsdt.bat.
4. Run CAT via desktop icon or Start Menu->Programs.

3. Introduction

Effects Based Operations is a core concept for the Advanced Technology Air Operations Center (AT AOC). AFRL's EBO ATD is a multi-year research and development initiative to develop software tools that support EBO. These tools focus on a range of activities, including Predictive Battlespace Awareness, effects-based COA development, and effects-based assessment both during planning and execution. Today's demonstration offers a brief glimpse into part of the ATD's research.

The Strategy Development Tool primarily supports Mission Analysis and Course of Action (COA) Development, the first two steps in the six-step Joint Air Estimate Process (JAEP). The SDT also provides tools for target set development and predictive nodal analysis, based on selected target system models. The Causal Analysis Tool, or CAT, supports JAEP's third step, COA Analysis, and is used during planning for an early assessment of the probability of achieving the objectives in the SDT's COA. During execution, it acts as a Campaign Assessment Tool, revising probability estimates based on combat assessment feedback from ISR assets. Based on this feedback, the COA in the SDT may be dynamically modified, or branch COA's may be selected according to the unfolding events in the battlespace.

This demonstration will focus mainly on the COA development and Analysis steps of the JAEP, and only briefly on Mission Analysis. In the demo, we will use the SDT to model enemy Centers of Gravity (COGs), author a phased COA containing objectives, effects, causal linkages, tasks, and indicators for a single blue plan. We will also perform a target set analysis in the SDT to predict the enemy's response to the outages incurred by the plan's targets. Then we will use CAT to give an assessment of the COA's probability of achieving commander's intent.

Operation Deny Force (ODF) serves as the challenge problem scenario for conducting experiments and demonstrations in support of EBO ATD and related AFRL efforts. A Joint Air Operations Plan, scenario description, and target system models based on the Pacifica dataset from JEFX '99 are provided as support to the technology developers. ODF is an abstraction loosely based on previous actual military operations. In this scenario, a "rogue" nation, Califon, is developing Weapons of Mass Destruction capabilities and the means to deliver them throughout the region, threatening destabilization and harm to US interests and assets in the region. The NCA directs the USMEDCOM to begin developing plans to deter Califon from deploying WMD systems, and if deterrence fails, to be prepared to disrupt or destroy those systems.

4. Plan Authoring

We'll begin the demonstration with a quick overview of SDT's mission analysis capability. In the interest of time, we will load a blank plan that contains pre-entered mission analysis information.

[Launch SDT. In the SDT main window, choose the Open Plan toolbar button (open folder icon).
In the file chooser, choose ODF_CAOC_MISSION_1.4.xml from the menu and click OK.]

The COA editor is divided into two panes. The left side is a tree view showing the hierarchical plan structure. The right side may be one of several graphical and text-based editors describing the currently selected plan element. The "View" toolbar buttons allow users to switch between these different planning tools, and contains several tools that run in their own windows, including a map tool, mission analysis dialog, and commander's intent dialog.

4.1 Campaign-level Plan Elements

4.1.1 Mission Analysis

Launching the Mission Analysis dialog reveals the information the JFACC's planning team has gleaned from the Joint Force Commander's warning order, which is a document containing the joint-level goals of the operation. Text from this document may be copied and pasted into fields in the SDT plan editor using standard Windows copy/paste shortcuts like Ctrl-C, Ctrl-V.

[Click on the "MA" button in the View toolbar]

Mission Analysis consists of five subtopics, each entered in a separate tab of this dialog. As an embedded help tool, the definition of the subtopic is shown by mousing over the word at the top of the pane.

[Click on the Desired Effects tab. Mouse over the word "Desired Effects" to show the definition as a tooltip popup]
1. Desired Effects provides text fields for entering top-level desired effects for the operation. Here, the primary desired effect is to deter WMD deployment. If this fails, a secondary desired effect is to disrupt or destroy WMD capabilities.
[Click on the Tasks tab. Mouse over the word "Tasks" to show the definition.]
2. The Tasks tab provides three areas for entering specified, implied, and essential tasks for the operation. The specified tasks here have been directly cut and pasted from the warning order document. The JFACC planning team will then derive any implied tasks and enter essential tasks as a subset of the specified and implied tasks.
[Click on the Resources tab. Mouse over the word "Resources" to show the definition]
3. Resources describes special resources that may be available for the operation.
[Click on the ROEs tab. Mouse over the word "ROE" to show the definition]
4. The Rules of Engagement tab provides two areas for entering constraints (required actions) and restraints (prohibited actions), which here include limitations on striking civilian infrastructure or political leadership.
[Click on the Mission tab.]
5. Finally, the Mission tab provides a single textfield for entering a concise mission statement describing the operation.
[Click Reset / Cancel to close the Mission Analysis dialog]

4.1.2 Commander's Intent

Next we will look at the JFACC Commander's Intent for Operation Deny Force.
[Click on the CI button in the View toolbar]
This describes the JFACC's desired endstate, purpose, prescribed methods, and allowable risk associated with the air component's plan. It is derived from the Mission Analysis information, and often is negotiated iteratively with the JFC and other component commanders.
[Click on the Purpose tab and invoke the mouseover definition]
1. The Purpose tab provides a single textfield for describing the purpose of the operation, in this case regional stability and reduced threat to US forces.
[Click on the Method tab and invoke the mouseover definition]
2. The Method tab provides guidance as to the means of carrying out the operation, in this case a mix of lethal and nonlethal strikes.
[Click on the Endstate tab and invoke the mouseover definition]
3. The Endstate tab provides multiple textfields describing how the military and political situation should appear at the end of the operation.
[Click on the Risk tab and invoke the mouseover definition]
4. Finally, the Risk tab provides multiple textfields describing the expected or acceptable risk factors of the operation. Here, the desired risk is low for US forces and for collateral damage to non-WMD enemy facilities.
[Click on Cancel to close the Commander's Intent dialog]

4.1.3 Plan

Before we begin COA development, let's look at the information fields provided at the plan level. The root node in SDT is always a plan, representing the air component of the operation.

[Click on the "Operation Deny Force" tree node. A set of tabbed panes for the Plan appear on the right side. Make sure the "Attributes" tab is selected.]

The Attributes tab provides a field for a short name of the plan, and a larger field for a description. It also allows users to specify if they are planning from the blue (friendly), red (enemy), or gray (neutral) perspective.

[Click on the Relations tab.]

The Relations tab is a read-only list of the COAs included under this plan. Currently there are none.

[Click on the Facts tab and invoke the mouseover definition]

The Facts tab provides multiple text fields for users to enter important facts to consider during COA development. The Assumptions tab provides similar fields for key assumptions.

[Click on the When tab]

The When tab allows users to define calendar origin dates for relative or "reference" date scales such as C-day/L-hour and D-day/H-hour.

[Uncheck "unspecified" for both C-day and D-day.]

C-day must be defined in terms of a calendar date. The default date is the current day, but these choice boxes allow a different date to be entered.

D-day may be specified in terms of either a calendar date or a C-day. Let's specify D-day as C+10 for example.

[Click on the "Reference" radio button next to D-day. Select "C", "+", "10" in the choice boxes. Click "Apply" to register this definition].

Finally, planners may add their own customized reference date scales via the Add Timeline button. They may specify if the scale uses days or hours as units, and may define the origin as a Calendar date, or a C-day or D-day reference date.

[Click on Add Timeline. Click on the "Reference" radio button next to the combo showing "E". Select "D", "+" , "10"]

4.1.4 COA

Next we will define the main COA for Operation Deny Force. SDT provides the ability to specify multiple COAs, but here we will only define one. The SDT's "Add" toolbar shows what types of plan elements may be legally added under the currently selected element. In this case, COA is the only possible type we may add.

[Click on the "C" button in the Add toolbar.]

Since there is only one COA in this plan, it is automatically designated as the selected COA for execution, as marked by the dot next to the C in the tree view icon. Let's fill in some of the attributes for the COA.

[On the right-hand side click on the Attributes tab. Type in the following, reading as you type:
Name: *Deter WMD – on order Disrupt/Destroy*
Description: *Deter Califon from using WMD. If deterrence fails, disrupt or distroy Califon's WMD capability.*

As before, a popup definition of the plan element being edited in the current tab appears if you hold the cursor over the attribute's label.

[Demonstrate by holding cursor over the "COA" label at the top of the dialog until popup text appears]

Clicking Apply saves our changes, while Reset will undo the changes and reset the editor to show the properties when we first clicked on the node.

[Click Apply to close the dialog]

4.1.5 Phase

Next let's define the phase structure of our COA using the Phase chart. This chart is only available when a COA is selected in the tree view.

[Click on the Phase view button in the View toolbar]

This editor allows users to enter an initial phase, a series of sequel phases, and branch phases that are conditioned on a triggering event. The ODF COA will consist of five main sequel phases, and one branch phase. The initial phase is Deployment.

[Click on the Initial Phase button at the bottom of the phase editor. Click on the phase box that appears. Right-click and select Edit. Type "Phase 1 – Deployment" in the Name field and click OK].

Phase 2 is Deterrence, which we add as a sequel to Phase 1.

[Click on the "Phase" button at the bottom of the phase editor. Click on the new phase box that appears. Right-click and select Edit. Type "Phase 2 – Deterrence" in the Name field and click OK].

We then add a sequel for Phase 3 to Disrupt WMD, and placeholder phases for compliance and redeployment, which we will not cover during this demo.

[Click on the "Phase" button at the bottom of the phase editor. Click on the new phase box that appears. Right-click and select Edit. Type "Phase 3 – Disrupt WMD" in the Name field and

click OK. Repeat this process for "Phase 4 – Compliance" and "Phase 5 – Redeployment."]

Now we also want a branch phase during Phase 3 to Destroy WMD on the condition that Califon is preparing to launch their missiles.

[Click on "Phase 3 – Disrupt WMD". Click on the "Phase" button at the bottom of the editor. Click on the new phase box that appears. Right-click and select Edit. Type "Phase 3 Branch – Destroy WMD" in the Name field, and type "Indications of WMD deployment" in the Branch Condition field and click OK].

To see the overall phase structure we can zoom to fit the view.

[Click on the Zoom to Fit button at the top of the editor].

To save time, we'll open a version of the plan that has the remaining attributes of phases already filled in.

[Choose File->Open, answer "No" to the prompt asking if you want to save changes, and select ODF_CAOC_PHASES_1.4.xml].

4.2 Objective-level Plan Elements

4.2.1 Objective

There are three ways to elaborate phases in SDT: manual entry, inserting a mission template, or exporting a causal chain from a COG model. We will cover all three in this demo. We'll start with manual entry to give an overview of each type of plan element. We'll skip ahead to Phase 3 Branch – Destroy WMD, and come back to the earlier phases to demonstrate the other two methods.

The SDT supports authoring of effects-based plans using a hierarchical plan ontology jointly developed by ISX Corporation and ALPHATECH. We have already covered the top three plan elements in this hierarchy: plans, COAs, and phases. The next level plan element is an objective, which represents a top-level desired goal within a COA or phase.

[Click on the phase "Phase 3 Branch – Destroy WMD" in the SDT tree view. Then right-click and select "New Objective" (checkered flag)].

To name and edit this objective, we invoke its properties dialog as we did for the earlier plan elements.

[In the right-hand side, mouse over the "Objective" title and read off the definition. Then type in the following attributes, reading them as you type:
Name: *Eliminate Califon WMD threat to region*
Description: *Disable Califon's WMD delivery capabilities].*

After typing the name and description, we may set the priority and weight of effort, either as a rank/percentage, or as qualitative labels.

[Choose the "Label" radio button next to Priority, and select "High" as the priority. Choose the "Label" radio button next to Weight of Effort and select "High"].

[Click on the Relations tab].

The Relations tab shows what parent phase or objective this objective supports, as well as child subobjectives and subeffects, which are currently blank.

[Click Apply to save the changes].

Note that a graphical view of the effects and tasks under the currently selected objective is available by switching to the Objectives editor.

[Click on the Objective editor toolbar button (second one in View group) to show the ISX Objectives editor. Click on the + zoom button to zoom in until the names of objects are visible]

This editor graphically displays the causal linkages between the objective's subeffects and tasks. We can also use this as an alternative to the SDT's "Add" toolbar for adding effects, linkages and tasks under an objective.

4.2.2 Effect

Now let's elaborate this objective with by entering desired effects, causal linkages, and tasks.

[Right-click on the "Eliminate…" objective in the tree view, and select "New Effect". In the right-hand side, mouse over the "Effect" title and read off the definition.]

We'll start by giving the effect a short name, "Disable WMD/TBM assets."

[Type "Disable WMD/TBM assets in the Name field].

Next we'll choose the relevant COG and effect verb. The choice of allowable effect verbs depends on the type of COG. For example, we can deceive leadership but not attrit it. We cannot deceive infrastructure but we can disrupt it. Choosing "general" as the COG type causes the entire suite of effect verbs to be available, including the standard "D11." The effect verb choicebox is editable, so users may override the provided choices and type in their own type of effect. For this effect, let's choose General as the COG and "disable" as the effect verb.

[Choose "Fielded Forces" for the COG and "disable" for the effect verb.]

Other effect attributes include:
- Indirect Object of Effect: This optional field refers to objects that are indirectly affected by the current affect. For example, "Disrupt EP supply for WMD facilities" has EP supply as the direct object of effect and WMD facilities as the indirect object of effect.
- Desired Level: This optional field represents a desired maximum threshold for the reduced probability or supply level of the object of effect. This threshold is compared to the predicted probability or level reported in CAT or Endstate.

- Priority and Weight of Effort are similar to what we saw for Objectives and Phases. They are provided at all levels of planning.

[Click on the Relations tab]

The Relations tab shows the objectives this effect supports, any downstream causal linkages it triggers, and any upstream causal linkages it is caused by. Optionally, a list of specific affected targets may be imported from the target set tool or added manually.

[Click on the "When" tab]

The When tab allows planners to specify desired timing of the effect. The start date for the effect may be entered as a calendar date or reference date. Here we choose C+15.

[Click on the Reference radio button under desired start date. Choose "C", "+", and click in the value combo and type in "15"].

We can also specify the desired overall duration of the effect, resulting from the cumulative persistence from multiple upstream causal links. Here we specify 15 days.

[Type 15 in the first textfield for Desired Duration. Choose "days" as the time units]

Finally we can specify the local delay and persistence for the effect. Here, persistence refers to the minimum duration of the effect resulting from a single upstream causal link, hence it is typically equal to or less than the previous desired duration.

[Type 15 in the first textfield for persistence and choose days. Type 0 for delay]

[Click on the Where pane]

The where pane simply allows users to type in named locations, TAIs, or NAIs where this effect should occur. A2IPB is one possible source for this information, which may be seen in the map tool.

[Type in "Califon" as the location.]

[Click on the MOEs pane]

The MOEs pane allows users to specify one or more measures of effectiveness for this Effect. By definition, an MOE must be a "standard" that has both a criterion and a measured element. The criterion is a constraint such as "less than" or "more than" along with a desired threshold value. The measured element is an observable piece of evidence that could be derived from ISR or HUMINT reports.

[Click on the first MOE listed at the top of the dialog to select it (it should highlight yellow). For "Object" type TBM launchers. For "Attribute" type Mission Readiness rate (%). Click on "Add Constraint". Under the Constraint column choose "<". Under the "Value" column click and type "20".]

Users may also optionally specify the desired timing for each constraint in terms of when it should be achieved and how long it should be maintained. This may be done in the last two columns of the constraints table.

[Click on the "Indicators" tab and Click "Add"]

Indicators are similar to MOEs, except that the criterion is optional, and an indicator may be either positive or negative evidence of achieving the effect.

[Click "Apply" to save changes]

4.2.3 Causal Linkage

Now that we've specified the main effect, we need to decide what type of causal linkage is used to achieve it.

[Right-click on the "Disable WMD/TBM Assets" Effect in the tree view, and choose "New Causal Linkage." In the right-hand side, mouse over the "Causal Linkage" title and read off the definition.]

Causal linkages are the glue between the tasks and effects that describe how the tasks achieve the effect. They may often be a generalization of the type of attack used in the tasks. Causal linkages may be kinetic or non-kinetic. Examples of kinetic linkages include aerial bombing, ground attack, or sabotage. Examples of non-kinetic linkages include jamming, disinformation, diplomacy, coercion, psyops, or other non-physical means of achieving effects. For this effect, we will use the causal linkage "Aerial Bombing."

[Type in "Aerial Bombing" in the Name field. Type in "Surgical strike by GSTF against WMD C2, R&D, storage and assembly" in the Description field.]

Optionally, users may enter a qualitative explanation or formal statement of how this causal linkage should achieve the effect and why it was chosen. The strength of the linkage may be estimated either as a qualitative label, such as "strongly or weakly" or a probability as a percentage.   This controls the estimation how strongly the child tasks or subeffects influence the probability of the parent effect, used in the Causal Analysis Tool.

[Choose "strongly" in the strength field]

Finally, if this particular causal linkage takes a significant time to prepare or execute, users may these as delay in days or hours. Users may enter the persistence of the linkage, that is the expected duration over which it will be maintained, as well. We will go with the default of 0 days for delay and 1 day for persistence.

[Click on the Relations tab]

The relations tab shows what tasks or effects trigger this causal linkage and what effects it causes. Indicators may also be specified for causal linkages, in the same way as for effects.

Causal linkages support decomposition of the parent effect into tasks, subeffects, or even objectives (to represent different echelons of levels of planning). Here we will go directly to tasks.

[Click "Apply" to save the changes.]

4.2.4 Task

Tasks represent blue actions that occur at the lowest levels of the COA, typically directed at a single target set or target category.

[Left-click on the "Aerial Bombing" causal linkage in the tree view. Right-click and choose "New Task". In the right-hand side, mouse over the "Task" title and read off the definition.]

The properties for tasks are similar to effects except that the task verb combo box contains a slightly different list of verbs representing possible blue actions.

[For Name, type "Destroy WMD R&D facilities". For task verb, choose "destroy". For object of task type "WMD R&D facilities"].

Also, there is a field for estimating the probability of success of the task, either as a qualitative label like "high" or "medium" or as a percentage.

[Choose "Percentage" and select "95". Click on the Relations tab].

The relations tab shows any causal linkages triggered by this task, and the list of direct targets associated with the task, currently empty.

[Click on the When tab.]

The "When" tab is for specifying the start date of the task, either as a calendar date or a relative day/hour, and the duration, delay, and persistence of the task.

[Click on the "Reference" radio button. Select "C+15". Type "15" in the duration field and select "days" as the time units.]

The Where, MOEs, and Indicators tabs are exactly the same as for effects.

[Click "Apply" to save the changes.]

4.2.5 Copy / Paste

Now we will add three similar "destroy" tasks for other target categories using SDT's copy/paste capability.

[Left-Click on the "Destroy WMD R&D facilities" task in the tree view. Right-click and choose "Copy". Left-click on the parent causal linkage "Aerial bombing". Right-click and choose "Paste Node".]

We'll paste two more copies of this node, to specify four target categories in all.

[Right-click on "Aerial bombing" and click Paste Node two more times. Left-click on the first copy to select it].

Since we want to use the same timing for each task, we simply need to edit the object of effect and name to reflect different target categories.

[Change the name to "Destroy WMD C2" and the object of effect to "WMD C2".]

[Click on the second copy and change the name to "Destroy WMD assembly facilities" and object of effect to "WMD assembly facilities"]
[Click on the third copy and change the name to "Destroy WMD storage facilities" and object of effect to "WMD storage facilities"]

4.2.6 <u>Target</u>

Optionally, the planner may manually specify targets under tasks or effects, though this is typically done via the target query tool against a target database. Targets under tasks represent "direct" targets that are the objects of an initial blue action. Targets under effects represent indirectly "affected" targets that are at least one level removed from the original action. Here we will add a target solely for the purposes of describing its properties.

[Right-click on the task "Destroy WMD storage facility" in the tree view and choose "New Target".]

The top half of the dialog contains properties common to all types of targets, such as name, description, and priority.

[Type "Silo S1" in the Name field.]

The properties in the bottom half change depending on the selected target type. SDT supports four main types of targets: generic, enemy facility, enemy unit, or friendly asset (used to describe defensive targets to protect).   Facilities and units are typically queried from the MIDB via the SDT's target query tool. Other types are typically manually entered or derived from supporting tools or databases.

[Click on the "facility" radio button]

The "facility" type features additional attributes such as BE number, category code, and category description.

[Click on the "Where" tab]

The "Where" tab allows users to specify the latitude and longitude of the target. Typically this information also is pulled from a target database.

[Click on "Apply" to save changes].

4.3   Mission Templates

Now that we have covered all plan element types featured in SDT, let's look at the second way in which phases may be elaborated, mission templates. To speed up the COA development process, the SDT provides a small library of doctrinal mission templates. These templates are reusable effects-based plan fragments which describe canonical AF missions such as *Offensive Counter-Air* or *Search and Rescue*. Users may insert one of these templates under an appropriate level of the COA.

[Left-click on "Phase 2 – Deterrence"].

Here, we will choose to plug in a mission template for gaining air superiority, to implement the "deterrence" phase. The goal is that this show of force will deter the enemy from deploying WMD.

Click on the phase node "Phase 2 – Deterrence" in the tree view. Click on the "Open" folder button in the Mission Templates toolbar. Select "air_superiority.xml" and click Open. A new Objective will appear at the end of the existing nodes in Phase 2].

This mission has one main objective "Gain and Maintain air superiority" and one main effect of "destroy IADS." Four subtasks are shown for different target categories within the IADS, and several target nodes, each representing a different MIDB cat code, are shown under the tasks. SDT uses a convention in which target nodes that lack a BE number but contain a category code imply that all targets of that category in the region specified by the parent task should be considered as targets.

[Click on the objective, effect, causal linkage, task, and targets in the mission template as you talk through the above paragraph to show the properties of each node on the right-hand pane.]

Note that this mission template was already completely filled out, for demo purposes. Most mission templates omit context-specific information such as the timing and location. Specific thresholds and timing for MOEs are also left out for the planner to complete.

To publish any portion of a plan as a template, simply select the desired node in the plan tree, and click the Save Mission Template button. This will save the selected node and all its children under the given template name.

4.4 Strategy Templates

Similarly, to encourage strategy reuse and reduce the planning cycle, the SDT also has a library of Strategy Templates, which are strategic-level plan structures.   We have built strategy templates for denial, paralysis, and attrition. The elements of a Strategy Template come from IPB research on the enemy system of interest. We will briefly show a strategy template for "paralysis". First let's save our changes to this plan.

[Choose File->Save As, and type "demo.xml". Click OK].

[Click on the Templates->Apply Strategy Template in the menubar. Choose the template "paralysis.xml". Ignore the "Unable to apply strategy template" warning]

[Describe key strategic elements of the template, and how they relate to the ODF plan]

This strategy template has already been populated with IPB elements, so it is ready for users to flesh out detail by adding lower-level plan elements. Other strategy templates specify only "types" of IPB elements rather than instances. The SDT's effects-based ontology constrains users to select only those types of IPB elements that are appropriate for the chosen effect.
1.
[If time permits, show an example of using the ontology server to find a situation element for an affected item by:
1. *Switch to the Objective View*
2. *Click on a Disrupt effect, then click the edit (pencil & paper icon) button.*
3. *If the effect is not of type Disrupt, choose Disrupt as the effect type.*
4. *Click on the "..." button next to the affected item*
5. *Browse to an appropriate situation element in the Situation Element tree view. For example MIDBEntity->Military Capability->WMDCapability->OrangelandWMDCapability.*
6. *Click Select. The affected item field should be updated with the choice. Click OK in the effect dialog to close it.]*

To apply a strategy template, the SDT's Template Application wizard walks planners through the process of choosing appropriate IPB instances from the IPB element browser. The result is a plan that is tailored to the current situation and country of interest. We will skip the template application process, in the interest of time.

Let's reload the Operation Deny Force plan we were working on before.

[Choose File->Open and type "demo.xml" and click OK. SDT should reload the plan that was saved previously]

4.5 Effect Dictionary

As a reference tool, the SDT includes an "effect dictionary" in which users can browse definitions of the D11 and other effect verbs and sample sentences as applied to various types of COGs or affected items.

[Click on the "?" toolbar button under the Plan button group].

The first choicebox is a list of possible effect verbs. The second is a list of types of affected items. The third is a list of the 5 Warden COG categories. Changing any of these choices changes the sample statement that appears. Sample statements may not be available for all combinations of affected item and COG. If a statement does not appear, try relaxing the affected item / COG choice to "general".

[Choose "general" (the last choice) under the "Affected Item" and "COG" menus].

This tool helps to reveal the subtle distinctions among effects such as disrupt vs. destroy (temporary vs. permanent incapacitation) vs. degrade (partial incapacitation).

[Choose "disrupt", "destroy", and "degrade" to reveal the definitions of these effects]

4.6 Timeline editor

As an alternative to the "When" tab in the Task editor dialog, the SDT includes a timeline editor, which displays Gantt bars in an MS-project-like fashion for each task. A rollup bar is displayed for each objective showing the earliest start date and latest end date of its subtasks. This view is only available for objective nodes or the root COA node.

NOTE: In version 1.4, you must run a different SDT launch icon to run with the Timeline editor enabled, due to incompatibilities with the COG Articulator's COTS dependencies. Save your plan, exit SDT via File->Exit, then rerun the "SDT-Timeline" icon, and reload the plan.

[Click on the objective node "Stop WMD activities..." in the tree view. Click on the Timeline toolbar button in the View group. Click on the blank (third) magnifying glass to fit the tasks to the view].

You may click and drag out one end of a task bar to change the C+day of its start or end time.

[Drag out one of the tasks' bars to extend its duration, then drag it back to where it originally was]

[Click on the "RNL" button in the View toolbar to return to the properties view].

5. Center of Gravity Modeling

Before we discuss the third way to elaborate phases, we need to introduce the COG Articulator, SDT's tool for modeling enemy Centers of Gravity. The first button in the Analysis toolbar launches the COG Articulator.

[Click on the first button (red network) in the Analysis toolbar. An empty COG Articulator window should appear].

This tool allows helps users "get inside the head" of the enemy and construct plans and models from the enemy's point of view showing the internal and cross-COG dependencies of various enemy systems. Once the models are constructed, the tool also helps users brainstorm and analyze options for intervening against these enemy systems, with the aid of the Causal Analysis Tool, or CAT.

5.1 Simple COG Model

We will demonstrate this process by building a very simple COG model, and then load a more complex model to continue the Operational Deny Force demo.

The COG Articulator represents causal models using two main building blocks: COG nodes and COG links. These two types may be further subdivided into blue COA nodes, red COG nodes,

causer links, and inhibitor links. Typically, the first step in the process is to build a COG model consisting only of red COG nodes inter-connected by causers. Then, overlay blue COA interventions with inhibitor links against lower-level red COG nodes and blue COA objective links inhibited by top-level red COG nodes.

5.1.1 Red COG Modeling

We'll begin by drawing a red COG node. Click and drag anywhere in the network view to do this.

[Click on the red circle in the toolbar. Drag out a red box in the middle of the network view].

After a node or link is added, the tool displays a properties dialog similar to those in SDT. Here we may specify the name, type, COG taxonomy and COG category. Node types include goal, belief, action, resource, or condition. COG taxonomies are domain-specific and include Warden's 5-ring COGs (Leadership, System Essentials, Infrastructure, Population, Fielded Forces) , Barlow's national elements of value, or the PMESII categories. Here we will use the name "Califon invades Nevidah," which is of type "action." For COG taxonomy we'll choose Warden, and for COG Category, Fielded Forces.

[Enter the above information into the properties panel].

The baseline probability shown here refers to the probability this action occurs without any outside influences. Let's set that low, at 0.2 (typically high values should only be entered for bottom-level nodes).

[Change the baseline probability to 0.2]

The delay and persistence have the same meaning as discussed earlier in the SDT plan editor. Delay refers to a wait interval before any immediate upstream influences are felt, and persistence refers to a minimum duration over which this node must remain true once it has become true. We will leave these at the default of 0 and 1 respectively.

[Click OK to close the dialog]

Now we will add a second red COG node at a lower level, "Support of political allies". For this node we will set a high baseline, because we estimate a high prior likelihood of this being true.

[Drag another red node underneath the first one. In the properties window, set the name to "Support of political allies". Set COG type to condition, COG taxonomy to Warden, COG Category to "Leadership". Set a baseline of 0.8. Leave delay, persistence as is].

Now we can describe the influence of the second node on the first via a causer link, which indicates a positive causal influence.

[Click on the causer button "+" on the toolbar. Click on the bottom node to establish the startpoint of the causer, then click on the top node to connect it.]

Links also have a name, but the type is determined by whether it was added as a causer or an inhibitor. Let's give this a name of "political confidence."

[Type "political confidence" in the link name]

All COG links have two main probability values. The strength determines how strongly the origin node increases the probability of the destination node when the origin node is true. Zero to one indicates an increase and zero to negative one indicates a decrease (an inhibitor). If it is one, the destination node will be guaranteed to be true if the origin node is true. If it is minus one, the destination node will be guaranteed to be false if the origin node is true. We'll set this to .75, indicating a strong influence.

[Set the strength to 0.75]

The leak strength is more subtle. This refers to how strongly the origin node increases or decreases the probability of the destination node when the origin is false. Positive one means the destination node will be guaranteed to be true if the origin is false, and minus one means the destination node is guaranteed to be false if the origin is false. Most of the time we will use 0 for this, which is the default. Delay and persistence will also be left at the default. The "Nodes" fields below are read-only and simply list the names of the origin and destination nodes on either end of the link.

[Click OK to close the dialog].

Now let's add an observable node. Observables represent indirect evidence of red COG node that are not themselves directly observable. They are typically pointed to by a causer link from a red COG node. In this case, we will define the observable "Ground forces cross border" as an indicator of "Califon attacks Nevidah."

[Click on the yellow oval in the toolbar. Drag out a box to the right of "Califon attacks Nevidah." In the properties window that appears, give it the name "Ground forces cross border." Give it a baseline probability of 0. Leave delay and persistence at 0, 1].

Now we will add a causer of full strength linking the red COG node to the observable.

[Click on the causer "+" button in the toolbar. Click on the red node "Califon attacks Nevidah" and then the observable that was just added. In the properties window that appears, give the link a strength of 1.0, and a leak strength of -1.0. Click OK to close the dialog.]

5.1.2 Blue Intervention Modeling

We have just built a very simple three-node COG model. Now we will shift to the blue point of view to illustrate options for disrupting red's goals. We will start by adding a blue objective to deter Califon's attack.

[Click on the blue oval in the toolbar. Drag a box above the node "Califon attacks Nevidah". Type in the name "Deter Califon attack"].

Blue objectives are modeled as having an inverse probability to one or more red COG nodes. To do this, we set the baseline probability zero, so that its probability is completely influenced by the COG node.

[Set the baseline probability to 0, leave the delay and persistence as is]

To set up an inverse relationship, we add an inhibitor and set the strength to -1 and the leak strength to 1.

[Click on the inhibitor "-" toolbar button. Click on the red COG node "Califon invades Nevidah" then click on the blue objective just added. In the properties window that comes up, set strength to -1 and leak strength to 1.]

Now that we've established our objective, we just need to add an intervention to disrupt the red system. We'll call this "Sanctions against allies."   The intervention will last 15 days so we will set a delay of 15.

[Click on the blue oval in the toolbar. Drag a box below both red COG nodes. Type in the name "Sanctions against allies." Set the persistence to 15.]

In this case, instead of specifying a baseline probability, which is by definition active at all times, we will schedule a probability of 1 at a specific time, time 0 in this case.

[Change the baseline probability to 0. Click on the Timed probabilities tab. Click on the "Add" button. In the new table entry that appears, click in the left column (time) and type "0". In the right column (probability), click and type "1".]

Now we will make this intervention inhibit the lower red COG node, "Support of political allies." We will give it a strength of -0.5, since sanctions in this region historically have a moderate probability of success. Leak strength will be 0, since the intervention should have no influence if it is false.

[Click on the inhibitor "-" toolbar button. Click from the intervention "Sanctions against allies" to the lower red COG node "Support of political allies". In the properties dialog, set the strength to -0.5 and leak strength to 0.]

We will also set a delay of 7 on this inhibitor, since sanctions take awhile to take affect. We will assume that once it takes affect, the effect will last at least ten days after sanctions have ended, so we set a persistence of 10.

[Set the delay on the inhibitor link to 7. Set persistence to 10].

5.1.3 COG Variables

So far, our model has been built specifically for a particular crisis, namely Califon's invasion. COG models may also be built much more generally by using variables in the names of the nodes. Such models can be considered "COG Templates" because they can be tailored to particular situations by simply assigning names to the variables. We will illustrate how this works for a single variable.

[Choose Nodes->Variables from the menubar]

Using the variables dialog, we can add variables of any user-defined name and type. Specifying a value is optional and is typically done later when tailoring a COG template. Here we will add a single variable, named "Leader L" of type "HeadOfState."

[Click on the "Add" button. In the row that appears, click in the Name column and type "Leader L" and press Enter. Click in the Type column and type "HeadOfState" and press Enter. Leave the value blank. Click Apply and then OK]

Once you have entered a type, it becomes available for use in new variables via a pull-down in the Type cell. Now let's add a node using this variable in the name. Adding the COG node works the same as before, but after typing in part of the name, we insert a variable name.

[Click on the red oval and drag a node to the left of the lower red COG node "Support of political allies". Erase the default name that appears (e.g. "Node1") and type "Terrorism sponsored by". Then Click on the "Insert" button. The name should now read "Terrorism sponsored by <Leader L>". Click OK]

COG nodes with unbound variables can easily be identified by the angle brackets around the variable name, in this case "Leader L". Just to make the context clearer, let's link this to the nearby red COG node using an inhibitor.

[Click on the inhibitor "-" button in the toolbar, click on "Terrorism sponsored by <Leader L>" and then "Support of political allies". Click OK in the dialog that appears].

Now we will set the variable value in the variables dialog.

[Choose Nodes->Variables from the menubar. Click in the Value column in row 1. Type in "Osama Hussein" and press Enter (to register the typing). Click Apply, then OK].

When we apply a name to a variable, it automatically updates the names of nodes containing that variable to show the new value rather than the bracketed variable name.

We've now covered all the main COG modeling features available in the COG articulator. Let's save this model and return to our Operation Deny Force scenario. Saving the model also automatically exports the model to a .cap file that can be loaded and analyzed in the Causal Analysis Tool. The COG Articulator has no analysis capability of its own – the CAT tool is the only way to see the actual probabilities over time resulting from the model.

[Choose File->Save Model As… from the menubar and type in "democog.xml". Click Save to close the dialog.]

5.2 Leadership Model

Now we will load a pre-existing COG model embodying Califon's Leadership Center of Gravity and show how the COG Articulator supports the third method of elaborating SDT objectives, by exporting causal chains.

[Choose File->Open and select the file "WMD_leadership_interventions.xml".]

The top-level red COG node in this model is Califon deployment of WMD. The rest of the red model describes what lower level actions, resources, conditions, and groups are necessary for this action, the relative strengths of their contributions, and the delays and persistences inherent in the process. The model is used in two ways. First, it is intended as a reference to inform the COA development process. Users may enter candidate blue objectives and interventions in the COG Articulator, export the entire model to CAT and inspect the resulting reduction in probability of Califon's actions. When the user decides on the best interventions, their causal chains may be exported to SDT's plan editor as plan fragments.

Second, the model is used internally by the SDT's target set analysis algorithms as a high-level resource allocation policy for Califon. The bottom nodes represent supply for a certain commodity by a certain group, such as "EP for Military." The interior nodes represent demand for a given commodity, weighted by the causal strength of the linkage from the supply node. The target set analysis algorithms use these relative demand priorities to predict the enemy's optimal redistribution of commodities in the face of the target outages chosen in the SDT COA.

Let's go over a few interesting details of this model. The enemy's main goal of "Deploy WMD" is broken into two conditions, ability to deploy, and willingness to deploy, two of the three cornerstones of Clausewitz' trinity theory (could, would, should). His willingness to deploy depends mainly on the support of allies and the military, and only very weakly on the population as shown by the thin causers labeled population. His ability to deploy depends on WMD delivery systems being operational, effective command and control, and availability of WMD itself. WMD availability in turn depends on WMD storage, which depends on R&D facilities being operational, which depends on ground LOCS, electric power, and raw materials. We have pre-specified three candidate blue interventions to destroy or disrupt each these three bottom-level resources. Each has been given a different delay and persistence, and in some cases, inhibitor strength based on studies of the relevant target systems. The delay represents how long we think the intervention will need to take effect. The persistence estimates the "workaround" time of the enemy in repairing or dealing with the problem. The inhibitor strength represents the likelihood the intervention will work, often based on past history.

5.2.1 Probabilistic COG Analysis using CAT

To predict which of these interventions will have the highest impact, we need to analyze the COG model in CAT. Whenever you save a model to XML in the COG Articulator, it automatically saves a CAT version of the model with a ".cap" extension. Let's load this model now.

[Start CAT. Choose File->Open. Browse to the SDT "models" directory, and open "WMD_leadership_interventions.cap".]

By default, the exported CAT model contains all the nodes, links, timing, and probability data entered into the COG model, but using CAT's native representation. In this example, to more clearly demonstrate analysis of alternate options, we have manually defined separate "COAs" for each of the three blue interventions, as well as a fourth "No Intervention" COA in which they are all turned off. The best way to compare results across multiple COAs is in CAT's analyzer view.

[Choose View->Analyze in the menubar. Drag the divider in the Analyzer split pane to the right so that the probability graph has more real estate]

Let's start with the "No Intervention" case.

[In the COA combo box in CAT's toolbar, select "No Interventions."]

To analyze each COA, we need to restart CAT's Bayes net analysis.

[Click on the Build Bayes Net button in CAT's floating toolbar. In the timing dialog, enter 50 for both time fields, and use "Feed Forward" as the sampling method. Click OK]

Then we simply double-click on the node we are interested in to see its probability profile.

[Double-click on "Deploy WMD". You should see a red line added to the graph display]

The red line shows the probability of red deploying WMD over time without any blue interventions. Because of a 7-day delay after R&D and a 3-day delay after Storage, it takes ten days for the enemy to reach its peak probability of 0.8.

[In the COA combo box select "EP". Click on the Build Bayes Net button on the floating toolbar. Keep the timing settings at 50 and sample method as "Feed Forward". Click OK. After the cursor returns and the network has recolored, double-click again on "Deploy WMD." You should see a new graph line of a different color].

With the EP intervention, we had specified a delay of 0 and a persistence of 5 days. Day 10 is the relevant time to consider since that's the first day the enemy would have WMD available anyway. We see that, compared to the No Intervention case, this option reduces red's probability of deploying WMD from .8 to .2, but it only lasts 5 days.

Let's look at the other two options now.

[Repeat the above steps for the "Ground LOCS" COA and the "Raw Materials" COA. You should see four graph lines in all when you are done]

The Ground LOCS option has a delay of 5 days, so it does not have any impact till day 15 at a probability of .5. It has a persistence of 10, so it lasts till day 25. The final option, Raw Materials, has a delay of 10 days, so it first impacts red on day 20 at probability .5. It has a persistence of 20 so it lasts till day 40.

Based on this analysis, disrupting EP has the strongest impact on red's probability of deployment. But it has the shortest duration of effect, so we would need to reschedule interventions frequently to sustain it. Based on this, we will choose EP as the option for the SDT COA. Let's exit CAT and return to the SDT.

[Choose File->Exit to close CAT. Bring the SDT plan editor window forward].

5.2.2 Exporting Causal Chains

The final step in our COG modeling demo is to elaborate an SDT phase based on interventions specified in the COG model. In our recent probabilistic analysis in CAT, we saw that EP had the strongest impact on the probability of the enemy deploying WMD. The COG Articulator allows us to import an entire chain in the COG model into an existing SDT COA as a plan fragment, saving time and effort on the planning side. The first step in this process is to select the root plan node in SDT under which we want to import the fragment. In this case, that is Phase 3 – Disrupt WMD.

[Left-click on the node "Phase 3 – Disrupt WMD" in the SDT tree view]

The next step is to highlight the relevant causal chain (path between two nodes) in the COG Articulator.

[Bring the COG Articulator window to the front. Click on "Disrupt EP substations". Shift-click on "Prevent Use of WMD". Right-click on that node and choose "Highlight." These two nodes and all nodes between them should show a darker background]

The last step is simply to perform the export.

[Right-click on "Prevent Use of WMD" and choose "Export". Bring SDT to the front]

We can see that a chain of nodes has been added under Phase 3 describing our intervention and its effects from the blue point of view. The Objective "Prevent use of WMD" is the highest-level node and the task "Disrupt EP substations" is the lowest-level. The red COG nodes in between were exported as "disrupt" effects acting on the object or action named by the red node. The COG links now show up as SDT causal linkages of the same name. Now that the COA structure has been automatically created, users only need to fill out properties that are SDT-specific such as priority, weight of effort, etc. As we refine this export capability, more attributes, such as task start dates and durations, will be automatically filled in from the information in the COG model.

This concludes the COG modeling and analysis part of the demonstration.

6. Target Set Analysis

Now that we have fleshed out the plan to the level of tasks, we need to identify target sets and perform a target set analysis. Again, for demo purposes, we focus on Phase 3, which is a disruption of the activities of research and development of WMD capability in Califon. Based on the leadership model's WMD production process and subject matter experts' advice, we chose to disrupt activity at a cluster of R&D facilities by disrupting the supply of EP to these facilities. As the SDT becomes more mature, this analysis and the visualization of potential choices will become more explicit, and will be assisted by software linked to the detailed target system models. The Leadership model explained earlier shows how we have captured some of this knowledge thus far.

Before we begin, let's skip to a version of the plan with timing details specified for all tasks.

[In SDT, choose File->Open and select ODF_CAOC_1.4.xml].

6.1 Map Tool

Let's look then at the information we have about target systems in Califon as we begin target set analysis.

 [Click on the Map Display button in the view toolbar (globe icon).]

This window shows the nodes and links of the networks available in the MIDB for target system analysis. Note that we are using an unclassified "enhanced" version of the MIDB that contains both nodes and links between producers and consumers in various target systems.

[Choose Help->Map Legend from the map menubar]

As shown by the legend, the light blue nodes represent EP transfer stations, the dark blue represent EP generators, the white nodes represent cities, and the green represent R&D facilities (nuclear, chem, bio, etc). In the POL/Natural Gas network, pipelines are purple, with terminal pink arrows showing likely supply links (either pipe or truck) for end consumers. Pink nodes are active POL/NG nodes (production, compression, distribution) while dark gray are POL consumers (military). We also can view LOC information such as roads and bridges, although this model is not used in the current analysis.

[Optional - Click on the Layers pulldown and check the Roads box. Check the Bridges box. Be sure to turn these off before proceding.]

[Zoom in by choosing Tools->Zoom In and dragging out a box around the cluster of nodes in the lower left part of the map (Los Angeles area).]

Through prior intelligence, we know that Califon's main cluster of WMD research facilities is located in the Los Angeles area.

[
1. *Click on MouseMode/Region in the map toolbar.*
2. *Drag a box around Los Angeles metropolitan area in SW Califon.*
3. *Right click on the region and then click "Show names" to see the names of all the facilities."]*

The white node is the city of Los Angeles, and the green nodes are nuclear and chem/bio R&D facilities arranged in a vertical cluster near the city. The north end of the cluster is approximately (34N, -118W) including the THOUSAND OAKS CHEM BIO RD, the PASADENA CHEM BIO RD and the LOS ANGELOS NUCLEAR R AND D.

6.2 Query Tool

Now we will use the Query Tool to guide target set selection and refinement. First we find the facilities that are to be affected by choosing appropriate target categories and location.

[Perform a query for chem/bio R&D facilities:

1. *Click on the "?" icon in the map tool*
2. *Ctrl-Click on the categories "Bio research" and "Nuclear research" to select both.*
3. *Click "Apply" to narrow the list of facilities in the results panel on the left.*
4. *Click on the "Location" tab.*
5. *Click "Apply" to use the sample area hard-coded for the demo. This narrows it down to a list of 10 facilities in the Area Of Interest (AOI). If the following coordinates do not appear, enter them manually – UL (34.2, -119.5), LR (33, -116).*
6. *Hit "Select All" at the bottom of the list and then "Select" in the upper toolbar of the query tool to see where these facilities are in the map. They will be highlighted with a yellow circle. Click "Show names" to see their names. To turn off names, choose Edit->Node Names->Hide all to turn them all off.]*

The map tool has two list boxes representing two types of target sets, one for directly targeted nodes, and one for indirectly affected nodes. Now that we have narrowed our list of targets, we register them as candidate affected nodes by clicking on the "Affect" button. Then we will associate the target set with the appropriate effect node in the SDT plan.

7. *Choose Edit->Affected Targets->Add Selected from the map menubar. The targets will appear in the "Affect Targets" panel at the lower right of the map window. Note - The facilities in the left panel in the Query Tool must still be selected for this action to work.*
8. *In the map window's Affected Target panel, right-click and choose Select All.*
9. *In the SDT, click on the effect node "Disrupt EP for R&D facilities" in the tree view.*
10. *Append the targets to the effect by clicking on the Target icon in the upper right of the map tool.*
11. *Choose Edit->Select Nodes->Deselect All.]*

The current "effect" targets are shown in the map with an orange ring and X, and now appear as target nodes in the SDT under the effect "Disrupt EP for R&D facilities."

6.3 Target Selection

Recall that we have chosen to disrupt the EP supply in 2 steps: degrading EP supplied by the national grid and disrupting EP from local (backup) generators. The first effect is achieved through a single task at C day +15 of attacking EP substations supplying the facilities. The second effect is achieved by two tasks: attacking the POL supply of the EP facilities at time C+16, and attacking the EP generators themselves at C+17. Let's use a "Links" query to find candidate EP targets to isolate the facilities for the task on C+15.

The SDT supports two methods of target selection. The user can manually select and enter targets into the plan. Alternatively, the Option Generator can analyze a partial plan and propose targets. We cover these two options below.

6.3.1 Manual Target Selection

In this method, the user determines which targets are chosen. This can be based on their domain knowledge or assisted by the Query Tool.

[Optional - Perform a query for EP nodes in the AOI that are no more than 3 links removed from the candidate R&D facilities.

1. *In the query tool, move to the Links tab.*
2. *Click Select All to select all of the previous WMD facilities in the panel on the left.*
3. *Click on the "Add Nodes" button to move these facilities to the Links list box.*
4. *Click "Reset query" in the upper left of the query tool*
5. *Enter 2 and "EP" in the Links query sentence*
6. *Select all and click "Apply" in the Links panel to narrow the list down to nodes within 2 links away from the WMD facilities*
7. *Go to the "Type" tab.*
8. *Ctrl-Click to select the EP substations and EP transfer stations categories.*
9. *Hit "Apply" to narrow the list down to 7 candidate targets within these categories.*
10. *Hit "Select All" in the lower left of the query tool to select all the resulting targets.*
11. *Hit "Select" in the upper toolbar of the query tool to see where these targets are on the map. Hit "Show names" to see their names.*

This time, we register this target set as a direct target set and associate it with a task in the SDT, rather than an effect.

1. *Choose Edit->Direct Targets->Add Selected in the map tool menubar. The targets should appear in the Direct Targets panel in the upper right of the map tool.*
2. *In the Direct Targets panel, right-click and choose Select All.*
3. *In the SDT, select the task "Attack EP substations" in the tree view.*
4. *In the map tool, click on the target icon to append the targets to that task.]*

The current "direct" targets are shown in the map with a red ring and crosshair +, and now appear as target nodes in the SDT under the task "Attack EP substations."

Similar queries would be used to find generators on the local network supplying the R&D facilities with EP and to locate the POL nodes providing them with fuel. These target sets would be associated with the tasks of "Attack EP backup" and "Attack POL for EP backup" respectively. Because of demo time limitations, we will skip these queries and load the final version of the plan with these additional targets.

6.3.2 Automatic Target Selection

In this mode, the Option Generator proposes targets. The user can confirm or reject the proposed targets. Confirmed targets are automatically added to the correct plan node.

Let's load a version of the plan that has only affected targets specified, and has no direct target sets specified under tasks.

[Choose File->Open, and select Option Generation Minimal.xml]

The "Option Generation" button launches the algorithms that look for the best candidate targets to achieve the specified effects against the indirect targets.

1. *In the SDT, click on the blue OG button.*
2. *In the Map Tool, select the Option Generation tab on the right.*

3. *After a minute or less, the background to the "Plan Root" text should turn yellow and an effect will appear under it.*
4. *Click on the plus sign next to the box to expand the tree.*
5. *Continue expanding until you expand a task node.*
6. *Under the task, the proposed target names are listed.*

The Option Generation tab shows the results using a shallow version of the SDT tree view that only shows effects, tasks and targets. The order of appearance should be the same as that in the SDT tree. The proposed direct target sets show up under the task nodes. Users may select one or more targets, approve or reject them, and add them to the SDT plan under the corresponding SDT task node.

[Determine whether to accept or reject the proposed targets]

7. *Click on a proposed target. In the map, the node will be selected and its name shown.*
8. *Right click on a target and select Confirm. The background of the target is now green.*
9. *Right click on a different target and select Reject. The background of the target is now red.*

The "Add to Plan" option sends targets to SDT in the same way as the target button in the map toolbar described earlier. But here it first prompts to clarify whether you want to send remaining unapproved targets and then remove this task from the OG panel.

[Add the confirmed targets to the plan.]

10. *Right click on the task node and select Add To Plan.*
11. *A dialog window will appear with two checkboxes. One asks if you want to accept the targets that you have neither confirmed nor rejected. Click No. The other asks if you want to delete this node and its descendents. Click Yes.*
12. *Only the effects that you have not processed remain in the tree.*
13. *In the SDT, the targets you accepted have been added to the appropriate node.*

After this, we can see in SDT that the accepted targets have been added to the appropriate task node.

6.4 Simulation Visualization

[In the SDT, load "ODF_CAOC_TARGETS_1.4.xml". This plan will have additional targets under the above effects and tasks for EP generators and POL supply.]

The next part of the demonstration shows how SDT's target-set analysis capability may be used to predict the impact and duration of outages in certain target systems such as EP and POL. Because this analysis takes from minutes to hours, to save time, we will show a playback of a previous target system analysis run on this plan's target sets using a simulation visualization layer on the map. This playback animation gives, for any particular time, a global view of the networks, their nodes and their predicted levels of production and supply to consumers. The time series record of these effects, so-called "outage profiles," are returned to the SDT COA

Editor via XML and can be displayed along with the plan once the Endstate simulation for target system analysis has completed. Let's load these outage profiles into the SDT now.

[In the SDT, click on the Target Systems analysis button (bulls-eye icon) in the Analysis toolbar. Two file choosers will appear, one after the other. In the first chooser, select "augmentation20030829.xml" and click Open. In the second chooser, select "tst-solutions20030829.dat" and click Open. After a pause, some of the target nodes in the SDT tree view should be highlighted yellow to indicate the results have been loaded].
2.
[Make the map view full-size.]

3.   Now we can inspect the status of the WMD facilities over time on the map. We need to enable the simulation state layer, named "Endstate."

[In the map tool go to the Layers menu, turn on the Endstate layer by selecting the menu item for it. Small bar graphs should appear next to the nodes in the map.]

The electric power consumed by each node is shown as a bar graph next to the node representing percent of normal EP consumption. The green bars represent POL consumption and the black and red bars represent EP consumption on nodes and links respectively. By moving the time slider, we can see how the electric power at the WMD facilities in the LA area dropped from full to zero at C+15. This is an effect of the task of bombing the EP substations. This analysis reveals that the subsequent tasks of bombing the POL supply and the local EP generators at C+16 and C+17 were not necessary since full disruption was achieved at C+15.

[Use the time slider (map must be maximized or enlarged horizontally for this to appear) in the upper right of the map tool to move from day 0 to day 15 and note how the EP graphs drop to zero. In particular, point out the nine R&D facilities chosen earlier, and how their EP supply goes from full at start time and drops dramatically at C+15.]

6.5 Outage Profile Graphs

We can also view an outage profile in the SDT for a single target or its parent effect as a line graph. Simply select one of the highlighted facility targets in the tree view and click on the Graph button. Again, this shows the drop in EP supply for this facility from full to 0 at C+15.

[In the SDT, select one of the highlighted R&D targets under "Disrupt EP for WMD R&D facilities" and click the SDT graph button in the analysis toolbar to show the entire outage profile .]

7. Causal Analysis

Now that we have shown the process for authoring the COA and Target System analysis, the next step is to export the COA to the Causal Analysis Tool to evaluate the probability of achieving commander's intent. This probability is updated in three stages. The first is based on the original COA alone, without targets. The next stage appends targets to tasks based on Target System analysis or queries from JTT and revises the probabilities based on target probability of

kill from weaponeering data. Finally, we run a scheduler to choose timing and weapons for each target and revise the probability time profile again.

To better illustrate this process, we will focus on the COA phases for which targets have not yet been chosen, namely Phase 2 and Phase 3's branch phase. We omit the main Phase 3 sequel because we do not have weaponeering data for the EP / POL targets previously chosen.

[Make sure CAT is running with a blank plan (File->New). Go to the SDT main window. Click the "Export COA for Causal Analysis" button. From the list of phases, choose Phase 2 and Phase 3 Branch and click OK. Quickly, move to the CAT window to see the plan build automatically in CAT]

7.1 Plan View

The SDT has just exported the COA to CAT as a dynamic Bayes network that captures the causal links between plan elements, and when they are active over time. To see this plan more clearly, let's invoke the auto-layout tool.

[Click on the Perform Layout button and choose "South" orientation and click OK to lay out the graph].

At the bottom level, we see each of the leaf level tasks from the ODF COA, colored with blue outlines. The downstream effects, objectives, and COA nodes toward the top are colored with red outlines.

[Click on examples of each type of node as it is described, to show its properties in the lower-left properties pane]

The arrows between tasks and effects represent causal causal linkages. The causal strength is derived from the discrete strength values we saw in the SDT's causal linkage explanation template, shown as H-value in the properties window at the lower left.

[Click on one or more causal linkages connecting a task and effect to show the different H values that appear in the properties pane]

7.2 Assessment View

Now that we have a Bayes net representation of the COA, we can analyze the probabilities of each of the nodes in the plan. Before we begin, we must save the plan to generate Bayes net files used by CAT.

[Choose File->Save from the menubar and save the plan. Choose a name that has not been used before in the given folder.]

Next, we start the Bayes net analysis and use the Hierarchy View for a rollup summary of probabilities.

[Click on the "Build Bayes Net" button in the CAT floating toolbar. In the "Timeline Scaling" dialog, make sure the "Simulation time slice count" is the same as the "estimated plan length" (use 50 for both), set sampling method to "Feed Forward" then click OK.

Select the "Hierarchy" tab in the lower-left side of CAT.   Double-click the "Default COA" folder to make sure all the nodes are listed below it. Double-click on the top-level node, "Operation Deny Force - PLAN" to see a "traffic light" icon showing its current probability. Click on the + next to this node and its children in the hierarchy view to expand the plan all the way down to tasks]

The nodes in this view are colored green for probabilities over 66%; yellow for 33-66%; red for 0-33%. At time 0, we have a low probability of achieving commander's intent because none of our tasks have executed yet. We can use the time slider to see this probability increase over time as the tasks trigger the interior effects in the Bayes net.

[Choose View->Time Selector to get the CAT time slider. Click on "Play" to animate the probability colors in the plan view and tree view]

This animation helps show that the IADS tasks occur first, followed by the WMD tasks. The COA achieves a maximum probability of 75% once both IADS and WMD effects have been achieved.

We can look at this "probability profile" in more detail by enabling the Analysis view.

[Click on "Done" in the Time Selector. Choose View->Analyze.   Scroll down to the PLAN node in the plan view and double-click on it. Also double-click on the two top-level effects "Loss of Air Sovereignty" and "Disable WMD/TBM Assets.]

This view shows the probability of the COA increases from 0 to 45% as the staggered IADS tasks occur over the first 15 days. Then it jumps to 75% when the two effects of Loss of Air Sovereignty and Disable WMD assets overlap from C+15 to C+30. Then it falls back down to 45% when the WMD effect expires.

This concludes our demonstration.

APPENDIX C

SDT PLAN REPRESENTATION AND XML SCHEMA

1. INTRODUCTION

The Strategy Development Tool (SDT) provides a suite of plan authoring and Course of Action (COA) analysis tools for joint air campaign planning. The SDT primarily supports Situation and COA development, the key second step in the Joint Air Estimate Process (JAEP). It transcends the capabilities of previous strategy-to-task planning tools by using adversarial models to guide choice of effects and actions at multiple levels of planning. The SDTs plan authoring tools use an effects-based plan ontology for decomposing the Commanders Guidance into appropriate phases, objectives, effects, causal linkages, tasks, and targets. The SDT also provides a COA Analysis capability, the third JAEP step, based on:

1.   Probabilistic analysis within AFRLs Causal Analysis Tool (CAT) to determine the probability of achieving the objectives in the COA over time.

2.   Strategic Center of Gravity (COG) models reflecting dependencies among the enemys goals, beliefs, and resources, used to assess the impact of interventions in the COA on the enemy systems.

3.   Target Systems Analysis to predict the enemy's response to destroyed targets or outages induced by the COA.

2. PRIMARY INPUTS AND OUTPUTS

SDT does not provide or export any specific interfaces that utilize the generated plan data files described in this document. Rather, SDT's primary input and output is an XML-based plan representation loaded and saved via the local file system. See Section 4 for a detailed description of SDT's XML plan schema.

3. SECONDARY INPUTS AND OUTPUTS

SDT is capable of sending and receiving plans via the Strategy Management Service (SMS). SMS acts as a plan repository that is accessed via a light weight SOAP (simple object access protocol) client that was implemented in Java. The SOAP client exports 5 methods that SDT uses, deletePlan, exportPlan, insertPlan, showListOfPlans, and showVersion. The signatures for each method can be found in Figure 50 to Figure 54. Figure 49 shows the exchange of data between SDT, SMS, and the SOAP client.. XML is used as the de facto data interchange format between the three tools.

Figure 49: SDT -> SMS work flow

deletePlan
```
public static java.lang.String deletePlan(java.lang.String soapUrl,
                                          java.lang.String fileName,
                                          java.lang.String userName,
                                          java.lang.String password)
```

- soapUrl: Path to DevNet = https://wlssvr.tbmcs-devnet.easistar.external.lmco.com/SISSVC/SISSVCWebService
- fileName: Name of plan file to be deleted
- userName: DevNet userName
- password: DevNet password

Returns:
SOAP response String

Figure 50: SMS DeletePlan method

DeletePlan is invoked when SDT wishes to delete a plan from the plan repository. The parameters are the url of the web service, file name to delete, and the user name and password. The SOAP client returns a string detailing if the operation was successful. This result is displayed to the user via a dialog box.

ExportPlan is used to download a plan from SMS. The web service takes four parameters-
the URL of the web service, the name of the SMS file to download, and the username and
password. The downloaded plan will be in the SMS XML format and thus is not directly
compatible with SDT's own plan representation. XSL (XML style sheet language,
http://www.w3.org/TR/xslt) scripts translate the SMS plan into one that is understandable to
SDT. This translation is invisible to the user.

exportPlan
```
public static java.lang.String exportPlan(java.lang.String soapUrl,
                                          java.lang.String fileName,
                                          java.lang.String userName,
                                          java.lang.String password)
```

- soapUrl: Path to DevNet = https://wlssvr.tbmcs-devnet.easistar.external.lmco.com/SISSVC/SISSVCWebService
- fileName: Name of plan file to be exported from server
- userName: DevNet userName
- password: DevNet password

Returns:
SOAP response String with plan xml

Figure 51: SMS ExportPlan method
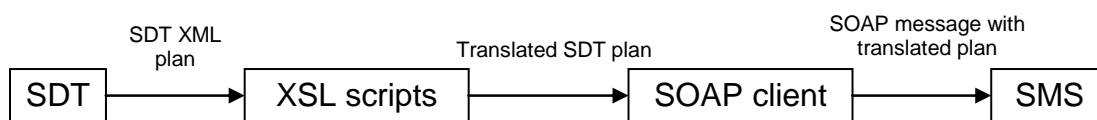
insertPlan
```
public static java.lang.String insertPlan(java.lang.String soapUrl,
                                          java.lang.String xmlFile,
                                          java.lang.String userName,
                                          java.lang.String password)
```

soapUrl: Path to DevNet = https://wlssvr.tbmcs-devnet.easistar.external.lmco.com/SISSVC/SISSVCWebService
xmlFile: Location of translated xml plan
userName: DevNet userName
password: DevNet password
Returns:
SOAP response String

Figure 52: SMS InsertPlan method

InsertPlan is used by SDT to upload a plan to the web service. This operation takes four
parameters: the URL of the web service, the name of the file that contains the SMS plan to

upload, and the user name and password. The response message is either a notification of success or an error message.

Before a plan can be exported to SMS using this web service, the plan must first be translated into the XML format for SMS, defined in the WSDL found at https://wlssvr.tbmcs-devnet.easistar.external.lmco.com/SISSVC/SISSVCWebService (DevNet login required). Using XSL translators, SDT converts its native XML plan representation into the format expected by the repository. This translation is invisible to the users.The SOAP client then invokes the remote SMS method to insert the plan into the SMS repository.

---

showVersion

```
public static java.lang.String showVersion(java.lang.String soapUrl,
                                            java.lang.String userName,
                                            java.lang.String password)
```

- soapUrl: Path to DevNet = https://wlssvr.tbmcs-devnet.easistar.external.lmco.com/SISSVC/SISSVCWebService

- userName: DevNet userName

- password: DevNet password

Returns:

SOAP response String naming SPT version on server

Figure 53: SMS ShowVersion method

The showVersion web service is used to retrieve the version of the SMS server.

---

showListOfPlans

```
public static java.lang.String showListOfPlans(java.lang.String soapUrl,
                                               java.lang.String userName,
                                               java.lang.String password)
```

- soapUrl: Path to DevNet = https://wlssvr.tbmcs-devnet.easistar.external.lmco.com/SISSVC/SISSVCWebService

- userName: DevNet userName

- password: DevNet password

Returns:

SOAP response String listing plans in database

Figure 54: SMS ShowListOfPlans method

ShowListOfPlans returns an XML document containing the list of plans that are currently stored in SMS. This list is parsed by SDT and displayed in a dialog that allows the end user to choose the plan to download.

4. SDT XML SCHEMA

This section presents a view of the elements that comprise SDT strategy plans. The key for each diagram is shown in Figure 56 in Appendix 5.2. Figure 55 shows the list of elements, complex and simple types, and groups contained in the SDT plans, where each of those types are well-defined XSD (XML Schema Definition) entities.

- Element: consists of pair of open and closed tags with any number of other elements nested, including text.
- Group: associates a name and optional annotations thereby allowing a type to easily incorporate all elements in the named group.
- Complex Type: an XML element that contains other elements and/or attributes.
- Simple Type: an XML element that can contain only text. It cannot contain any other elements or attributes

Elements	Groups	Complex types	Simple types
componentPlan	**effectVerbGroup**	**Action**	**CalendarDate**
	taskVerbGroup	**CausalLinkage**	**CalendarOrReferenceDate**
		Coa	**EffectType**
		CommandersIntent	**LevelOfWar**
		Criterion	**PointOfViewType**
		DNAListNode	**ReferenceDate**
		DNSListNode	**ReferenceDateTimeUnitType**
		EboEntity	**TaskType**
		Effect	**TemporalConstraintType**
		EffectRestrictionNode	**TimeUnit**
		MIDBTarget	
		MissionAnalysis	
		MissionAnalysisTask	
		Objective	
		Phase	
		Plan	
		PlanningDuration	
		ReferenceDateDefinition	
		RestrictionConstraint	
		RulesOfEngagement	
		SituationElement	
		Standard	
		Target	
		Task	
		TemporalConstraint	
		UITargetNode	

Figure 55: List of entities in SDT's plan format

4.1. ELEMENTS

4.1.1. Component Plan

diagram
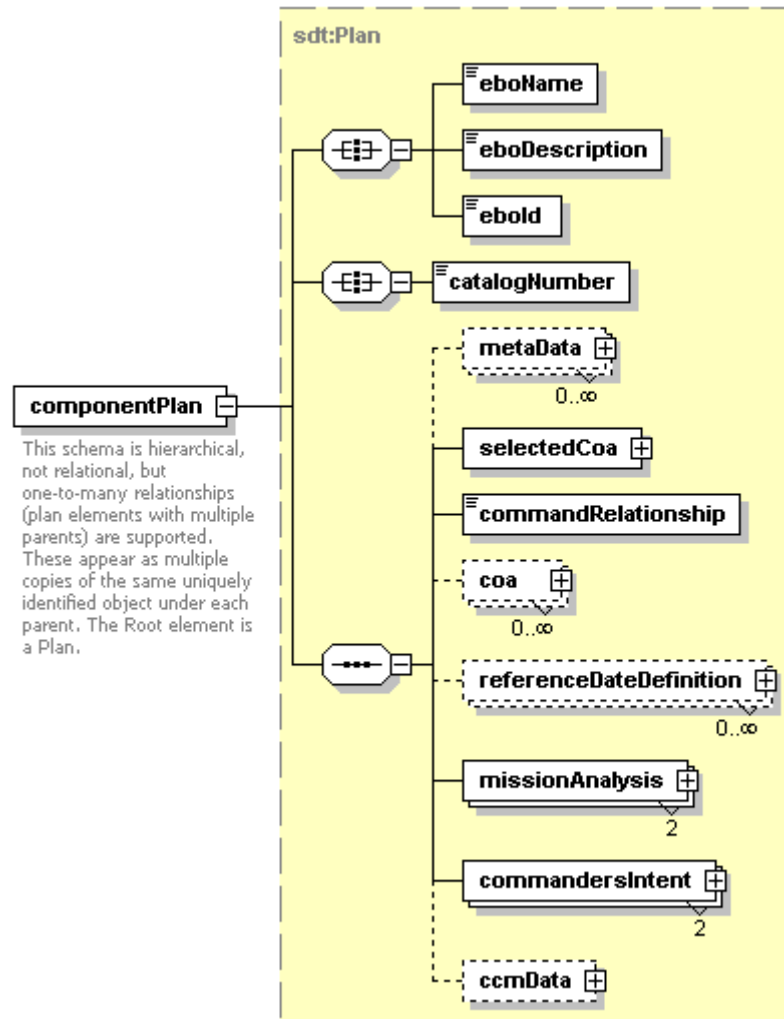
type	sdt:Plan
annotation	documentation This schema is hierarchical, not relational, but one-to-many relationships (plan elements with multiple parents) are supported. These appear as multiple copies of the same uniquely identified object under each parent. The Root element is a Plan.

source

```
<xsd:element name="componentPlan" type="Plan">

    <xsd:annotation>

        <xsd:documentation>This schema is hierarchical, not relational, but one-to-many
relationships (plan elements with multiple parents) are supported.  These appear as
multiple copies of the same uniquely identified object under each parent. The Root
element is a Plan.</xsd:documentation>

    </xsd:annotation>

</xsd:element>
```

4.2. GROUPS

4.2.1. Effect Verb Group

diagram
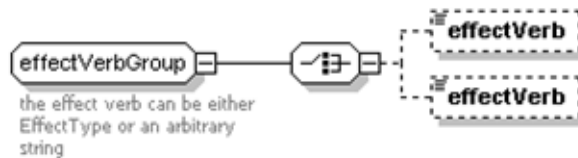
used by	complexType	**Effect**
annotation	documentation	the effect verb can be either EffectType or an arbitrary string

source

```xml
<xsd:group name="effectVerbGroup">
 <xsd:annotation>
  <xsd:documentation>the effect verb can be either EffectType or an arbitrary string</xsd:documentation>
 </xsd:annotation>
 <xsd:choice>
  <xsd:element name="effectVerb" type="sdt:EffectType" minOccurs="0"/>
  <xsd:element name="effectVerb" type="xsd:string" minOccurs="0"/>
 </xsd:choice>
</xsd:group>
```

4.2.2. Task Verb Group

diagram

used by	complexType	**Task**
annotation	documentation	the task verb can be either TaskType or an arbitrary string

source

```xml
<xsd:group name="taskVerbGroup">
 <xsd:annotation>
  <xsd:documentation>the task verb can be either TaskType or an arbitrary string
        </xsd:documentation>
 </xsd:annotation>
 <xsd:choice>
  <xsd:element name="taskVerb" type="sdt:TaskType" minOccurs="0"/>
  <xsd:element name="taskVerb" type="xsd:string" minOccurs="0"/>
 </xsd:choice>
</xsd:group>
```

4.3. COMPLEX TYPES

4.3.1. Action

diagram
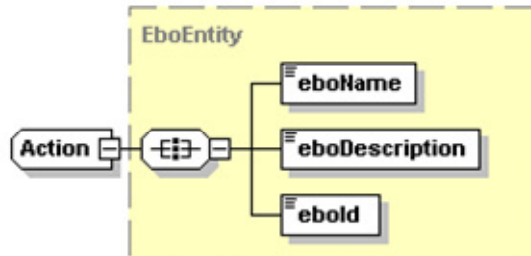
type	extension of **EboEntity**
used by	complexTypes **Objective** **Task**

source
```xsd
<xsd:complexType name="Action">
  <xsd:complexContent>
    <xsd:extension base="EboEntity"/>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.2. CausalLinkage

diagram

type   extension of **EboEntity**

used by   element   **Effect/causalLinkage**

annotation   documentation   CausalLinkage nests effects, tasks, and objectives.

source

```xml
<xsd:complexType name="CausalLinkage">
  <xsd:annotation>
    <xsd:documentation>CausalLinkage nests effects, tasks, and objectives.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="mechanismStatement" type="xsd:string" minOccurs="0"/>
        <xsd:element name="mechanismExplanation" type="xsd:string" minOccurs="0"/>
        <xsd:element name="indicator" type="sdt:Standard" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="effectTrigger" type="sdt:Effect" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="taskTrigger" type="sdt:Task" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="objectiveTrigger" type="sdt:Objective" minOccurs="0"
maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```
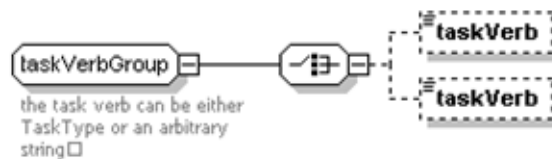
4.3.3. COA

diagram

type	extension of **EboEntity**
used by	elements **Plan/coa Plan/selectedCoa**
annotation	documentation Coas have any number of objectives, dns and dna lists, and at most 1 phase nested. DnaLis of "Do Not Affect" Effects. DnsList is the list of "Do Not Strike" targets.

source

```xsd
<xsd:complexType name="Coa">
  <xsd:annotation>
    <xsd:documentation>Coas have any number of objectives, dns and dna lists, and at most 1 phase nested.
DnaList is the list of "Do Not Affect" Effects. DnsList is the list of "Do Not Strike"
targets.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="objective" type="sdt:Objective" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="initialPhase" type="sdt:Phase" minOccurs="0"/>
        <xsd:element name="dnaList" type="sdt:DNAListNode" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="dnsList" type="sdt:DNSListNode" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.4. COA Comparison Node

diagram
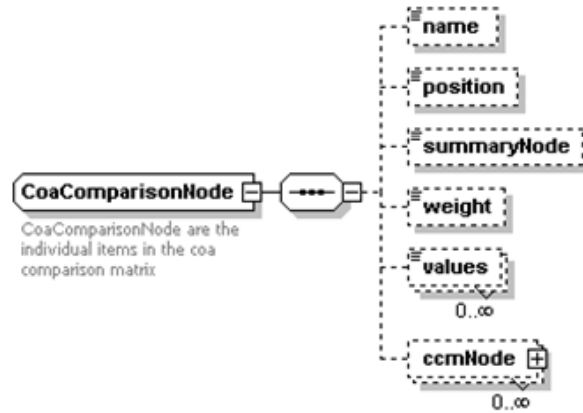
namespace	urn:com-alphatech-ebo:sdt:coa
annotation	documentation CoaComparisonNode are the individual items in the coa comparison matrix

source

```xml
<xsd:complexType name="CoaComparisonNode">
 <xsd:annotation>
  <xsd:documentation>CoaComparisonNode are the individual items in the coa comparison matrix</xsd:documentation>
 </xsd:annotation>
 <xsd:sequence>
  <xsd:element name="name" type="xsd:string" minOccurs="0"/>
  <xsd:element name="position" type="xsd:string" minOccurs="0"/>
  <xsd:element name="summaryNode" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="weight" type="xsd:decimal" minOccurs="0"/>
  <xsd:element name="values" type="xsd:decimal" minOccurs="0" maxOccurs="unbounded"/>
  <xsd:element name="ccmNode" type="sdt:CoaComparisonNode" minOccurs="0" maxOccurs="unbounded"/>
 </xsd:sequence>
</xsd:complexType>
```

4.3.5. COA Comparison Type

diagram
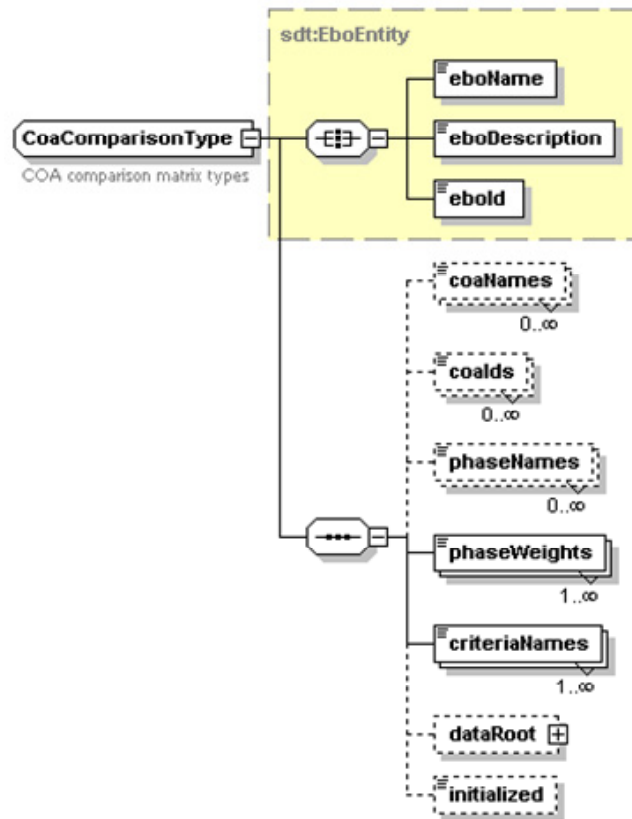
type	extension of sdt:EboEntity
annotation	documentation COA comparison matrix types

source

```xsd
<xsd:complexType name="CoaComparisonType">
  <xsd:annotation>
    <xsd:documentation>COA comparison matrix types</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="coaNames" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="coaIds" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="phaseNames" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="phaseWeights" type="xsd:decimal" maxOccurs="unbounded"/>
        <xsd:element name="criteriaNames" type="xsd:string" maxOccurs="unbounded"/>
        <xsd:element name="dataRoot" type="sdt:CoaComparisonNode" minOccurs="0"/>
        <xsd:element name="initialized" type="xsd:boolean" minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.6. CommandersIntent

type	extension of **EboEntity**
used by	element **Plan/commandersIntent**
annotation	documentation Commanders Intent is similar to MissionAnalysis
source	

```xml
<xsd:complexType name="CommandersIntent">
  <xsd:annotation>
    <xsd:documentation>Commanders Intent is similar to MissionAnalysis</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="endstate" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="purpose" type="xsd:string" minOccurs="0"/>
        <xsd:element name="method" type="xsd:string" minOccurs="0"/>
        <xsd:element name="risk" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.7. Criterion

diagram

type	extension of **EboEntity**	
used by	element	**Standard/criterion**

source

```
<xsd:complexType name="Criterion">
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="status" type="xsd:string" minOccurs="0"/>
        <xsd:element name="desiredValue" type="xsd:string" minOccurs="0"/>
        <xsd:element name="type" type="xsd:string" minOccurs="0"/>
        <xsd:element name="rangeConstraint" type="xsd:string" minOccurs="0"/>
        <xsd:element name="temporalConstraint" type="sdt:TemporalConstraint" minOccurs="0"
maxOccurs="unbounded"/>
        <xsd:element name="date" type="sdt:CalendarOrReferenceDate" minOccurs="0"/>
        <xsd:element name="constraintType" type="sdt:TemporalConstraint"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```
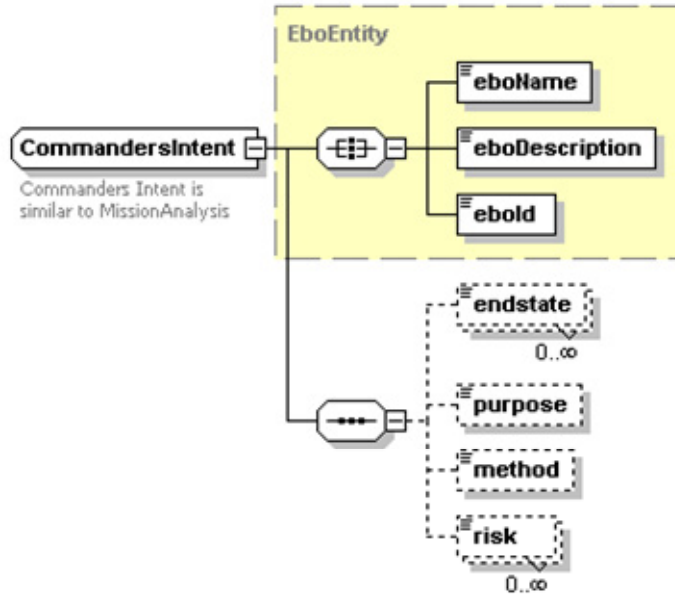
4.3.8. DNAListNode

type	extension of **EboEntity**
used by	element **Coa/dnaList**
annotation	documentation Contains a list of EffectRestriction elements.
source	

```
<xsd:complexType name="DNAListNode">
  <xsd:annotation>
    <xsd:documentation>Contains a list of EffectRestriction elements.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="restriction" type="sdt:EffectRestrictionNode" minOccurs="0"
maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.9. DNSListNode

diagram

type   extension of **EboEntity**

Used by   element   **Coa/dnsList**

annotation   documentation   Contains a list of Targets that should not be struck.

source
```xsd
<xsd:complexType name="DNSListNode">
  <xsd:annotation>
    <xsd:documentation>Contains a list of Targets that should not be struck.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="ueTarget" type="sdt:UITargetNode" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.10.   EboEntity

complexTypes **Action CausalLinkage Coa CommandersIntent Criterion DNAListNode DNSListNode Effect EffectRestrictionNode MissionAnalysis MissionAnalysisTask Phase Plan PlanningDuration ReferenceDateDefinition RestrictionConstraint RulesOfEngagement SituationElement Standard TemporalConstraint**

documentation EboEntity is the base class for all plan types. The eboId is unique within the context of the p necessarily in the context of all plans.

source

```
<xsd:complexType name="EboEntity">
 <xsd:annotation>
  <xsd:documentation>EboEntity is the base class for all plan types. The eboId is unique within the context
of the plan, but not necessarily in the context of all plans.</xsd:documentation>
 </xsd:annotation>
 <xsd:all>
  <xsd:element name="eboName" type="xsd:string"/>
  <xsd:element name="eboDescription" type="xsd:string"/>
  <xsd:element name="eboId" type="xsd:string"/>
 </xsd:all>
</xsd:>
```

4.3.11.　　Effect

diagram

157

type	extension of **EboEntity**
used by	elements **CausalLinkage/effectTrigger Objective/purpose**
annotation	documentation Effect has any number of causalLinkage and targets nested. Priority and weight may be eith numeric value or a text label.
source	

```xml
<xsd:complexType name="Effect">
  <xsd:annotation>
    <xsd:documentation>Effect has any number of causalLinkage and targets nested. Priority and weight may
be either a numeric value or a text label.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="priority" type="xsd:string" minOccurs="0"/>
        <xsd:element name="weight" type="xsd:string" minOccurs="0"/>
        <xsd:element name="purposeExplanation" type="xsd:string" minOccurs="0"/>
        <xsd:group ref="effectVerbGroup"/>
        <xsd:element name="objectOfEffect" type="sdt:SituationElement" minOccurs="0"/>
        <xsd:element name="indirectObjectOfEffect" type="sdt:SituationElement" minOccurs="0"/>
        <xsd:element name="locationOfEffect" type="sdt:SituationElement" minOccurs="0"
maxOccurs="unbounded"/>
        <xsd:element name="measureOfEffectiveness" type="sdt:Standard" minOccurs="0"
maxOccurs="unbounded"/>
        <xsd:element name="indicator" type="sdt:Standard" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="desiredEffect" type="sdt:Standard" minOccurs="0"/>
        <xsd:element name="causalLinkage" type="sdt:CausalLinkage" maxOccurs="unbounded"/>
        <xsd:element name="desiredStartDate" type="sdt:CalendarOrReferenceDate" minOccurs="0"/>
        <xsd:element name="desiredDuration" type="sdt:PlanningDuration" minOccurs="0"/>
        <xsd:element name="desiredLevelOfEffect" type="xsd:string" minOccurs="0"/>
        <xsd:element name="desiredLevelUnits" type="xsd:string" minOccurs="0"/>
        <xsd:element name="movementType" type="xsd:string" minOccurs="0"/>
        <xsd:choice>
          <xsd:element name="target" type="sdt:Target" minOccurs="0" maxOccurs="unbounded"/>
          <xsd:element name="target" type="sdt:MIDBTarget" minOccurs="0" maxOccurs="unbounded"/>
        </xsd:choice>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```
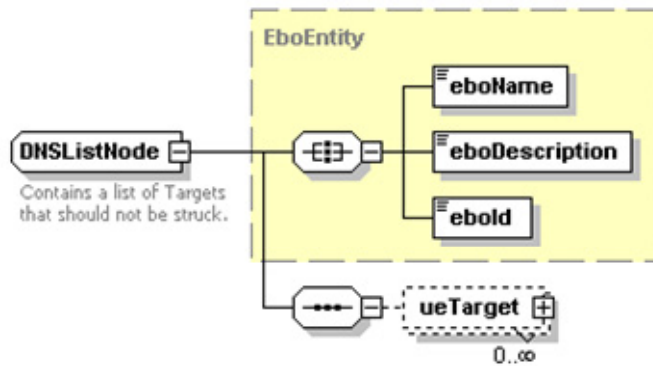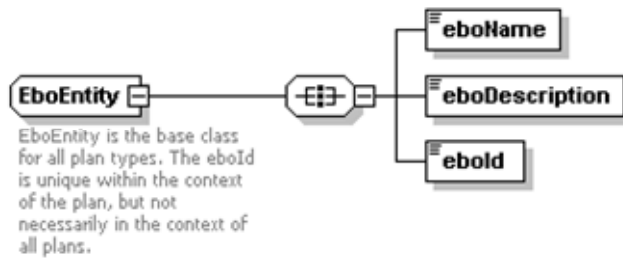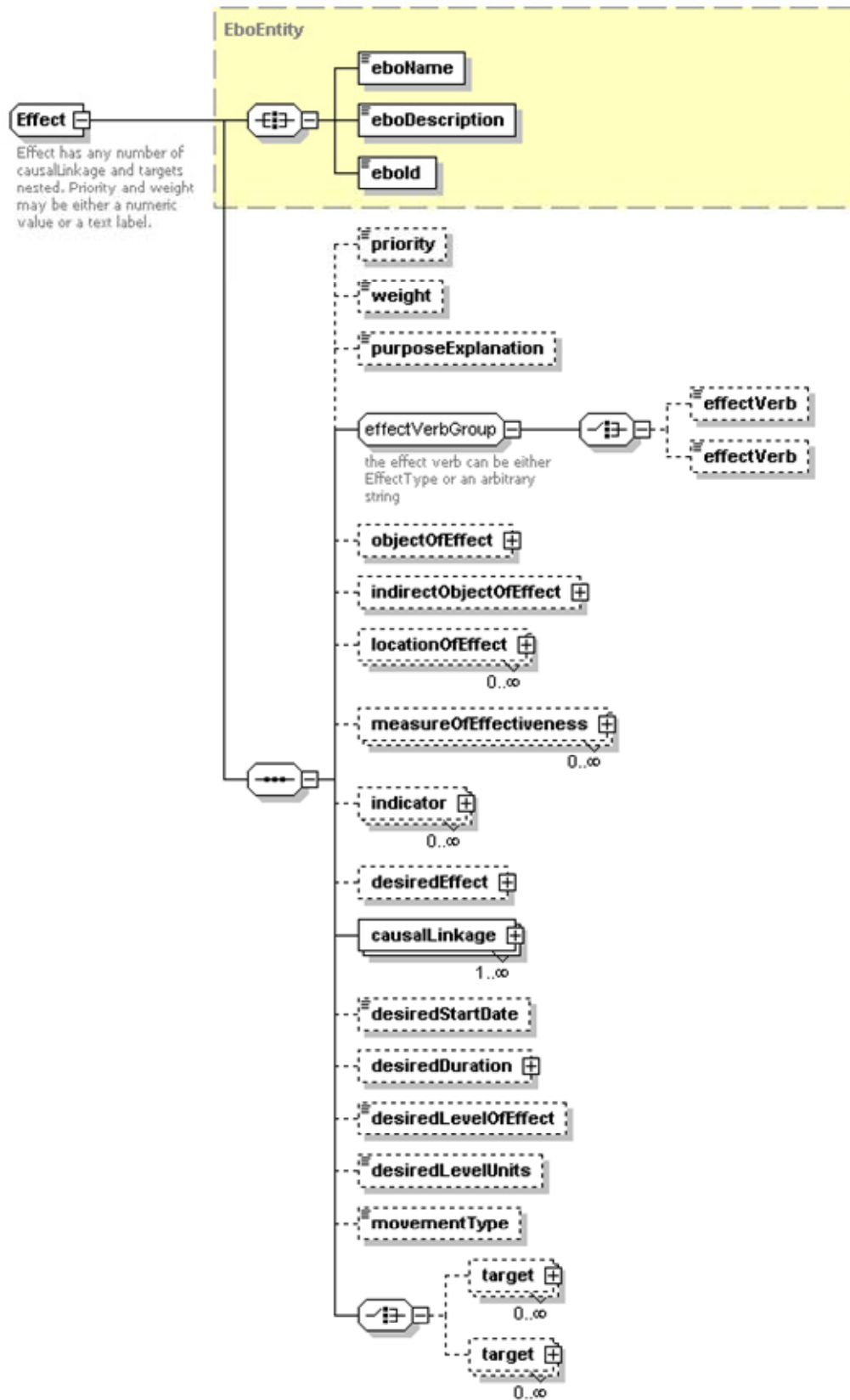
4.3.12. EffectRestrictionNode

diagram

type   extension of **EboEntity**

used by   element   **DNAListNode/restriction**

annotation   documentation   EffectRestriction has a list of Unintended Targets and constraints.

source
```xsd
<xsd:complexType name="EffectRestrictionNode">
 <xsd:annotation>
  <xsd:documentation>EffectRestriction has a list of Unintended Targets and
constraints.</xsd:documentation>
 </xsd:annotation>
 <xsd:complexContent>
  <xsd:extension base="EboEntity">
   <xsd:sequence>
    <xsd:element name="ueTarget" type="sdt:UITargetNode" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="constraint" type="sdt:RestrictionConstraint" minOccurs="0"
maxOccurs="unbounded"/>
   </xsd:sequence>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>
```

159

4.3.13.　MIDBTarget

diagram

type	extension of **Target**	
used by	elements	**Effect/target Task/target**
	complexType	**UITargetNode**
	documentation	MIDB targets extend Targets to provide additional information. The latitude and longitude a expressed in decimal format.
annotation		

source
```
<xsd:complexType name="MIDBTarget">
  <xsd:annotation>
    <xsd:documentation>MIDB targets extend Targets to provide additional information.  The latitude and
longitude are expressed in decimal format.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="Target">
      <xsd:all>
        <xsd:element name="beNumber" type="xsd:string" minOccurs="0"/>
        <xsd:element name="categoryCode" type="xsd:string" minOccurs="0"/>
        <xsd:element name="categoryDescription" type="xsd:string" minOccurs="0"/>
        <xsd:element name="latitude" type="xsd:decimal" minOccurs="0"/>
        <xsd:element name="longitude" type="xsd:decimal" minOccurs="0"/>
        <!-- Currently assumes decimal format -->
      </xsd:all>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.14. MissionAnalysis

diagram
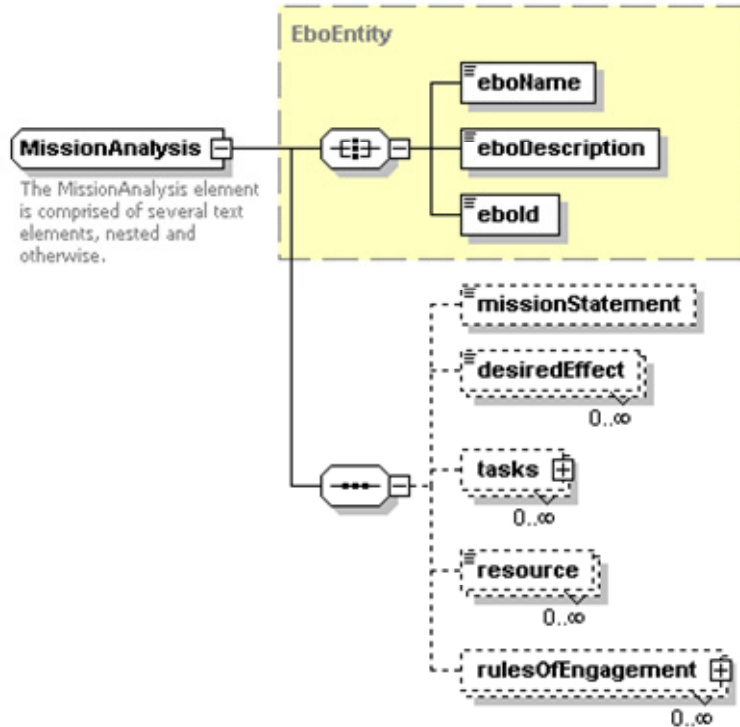
type   extension of **EboEntity**

used by   element   **Plan/missionAnalysis**

annotation   documentation   The MissionAnalysis element is comprised of several text elements, nested and otherwise.

source
```xml
<xsd:complexType name="MissionAnalysis">
  <xsd:annotation>
    <xsd:documentation>The MissionAnalysis element is comprised of several text elements, nested and
otherwise.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="missionStatement" type="xsd:string" minOccurs="0"/>
        <xsd:element name="desiredEffect" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="tasks" type="sdt:MissionAnalysisTask" minOccurs="0"
maxOccurs="unbounded"/>
        <xsd:element name="resource" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="rulesOfEngagement" type="sdt:RulesOfEngagement" minOccurs="0"
maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.15. MissionAnalysisTask

diagram

type   extension of **EboEntity**

used by   element   **MissionAnalysis/tasks**

annotation   documentation   MissionAnalysis Task provides 3 types of text areas to add tasks related to the mission anal

source
```xml
<xsd:complexType name="MissionAnalysisTask">
  <xsd:annotation>
    <xsd:documentation>MissionAnalysis Task provides 3 types of text areas to add tasks related to the
mission analysis.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="specifiedTask" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="impliedTask" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="essentialTask" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.16.   Objective

diagram

type   extension of **Action**

used by   elements   **Coa/objective Phase/objective Objective/objective CausalLinkage/objectiveTrigger**

annotation   documentation   Objective has any number of subobjectives and effects nested.

source
```xsd
<xsd:complexType name="Objective">
 <xsd:annotation>
  <xsd:documentation>Objective has any number of subobjectives and effects nested.</xsd:documentation>
 </xsd:annotation>
 <xsd:complexContent>
  <xsd:extension base="Action">
   <xsd:sequence>
    <xsd:element name="priority" type="xsd:string" minOccurs="0"/>
    <xsd:element name="weight" type="xsd:string" minOccurs="0"/>
    <xsd:element name="levelOfWar" type="sdt:LevelOfWar" minOccurs="0"/>
    <xsd:element name="objective" type="sdt:Objective" maxOccurs="unbounded"/>
    <xsd:element name="purpose" type="sdt:Effect" maxOccurs="unbounded"/>
   </xsd:sequence>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>
```

4.3.17. Phase

diagram

type	extension of **EboEntity**	
used by	elements	**Phase/branch Coa/initialPhase Phase/sequel**
annotation	documentation	Phase has 0 or 1 sequel Phases and any number of branch phases and objectives. The branchCondition says "How did I get to this phase?" and not "How do I choose which phase next?"

source

```xsd
<xsd:complexType name="Phase">
  <xsd:annotation>
    <xsd:documentation>Phase has 0 or 1 sequel Phases and any number of branch phases and objectives.
The branchCondition says "How did I get to this phase?" and not "How do I choose which phase to go to
next?"     </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="sequel" type="sdt:Phase" minOccurs="0"/>
        <xsd:element name="branch" type="sdt:Phase" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="objective" type="sdt:Objective" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="branchCondition" type="xsd:string" minOccurs="0"/>
        <xsd:element name="priority" type="xsd:string" minOccurs="0"/>
        <xsd:element name="weight" type="xsd:string" minOccurs="0"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```
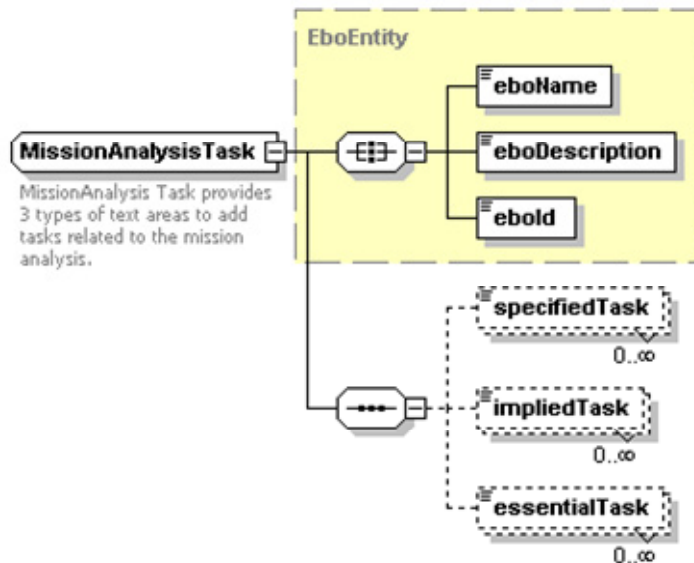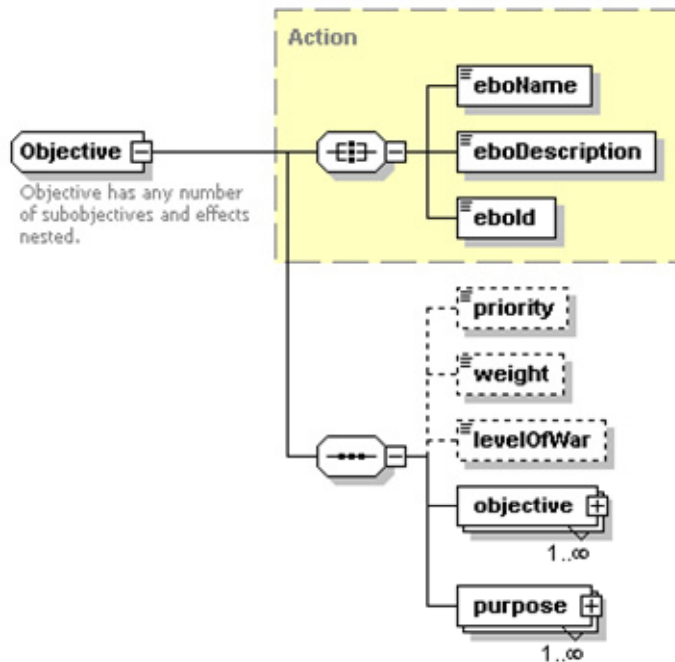
4.3.18.　　Plan

Each plan has 1 selected Coa and 0 or more alternate Coas.

type	extension of **EboEntity**
used by	element **componentPlan**
annotation	documentation	Each plan has 1 selected Coa and 0 or more alternate Coas.

source

```
<xsd:complexType name="Plan">
 <xsd:annotation>
  <xsd:documentation>Each plan has 1 selected Coa and 0 or more alternate Coas.</xsd:documentation>
 </xsd:annotation>
 <xsd:complexContent>
  <xsd:extension base="EboEntity">
   <xsd:sequence>
    <xsd:element name="selectedCoa" type="sdt:Coa"/>
    <xsd:element name="coa" type="sdt:Coa" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="fact" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="assumption" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="referenceDateDefinition" type="sdt:ReferenceDateDefinition" minOccurs="0"
maxOccurs="unbounded"/>
    <xsd:element name="pointOfView" type="sdt:PointOfViewType" minOccurs="0"/>
    <xsd:element name="missionAnalysis" type="sdt:MissionAnalysis" minOccurs="0"/>
    <xsd:element name="commandersIntent" type="sdt:CommandersIntent" minOccurs="0"/>
   </xsd:sequence>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>
```
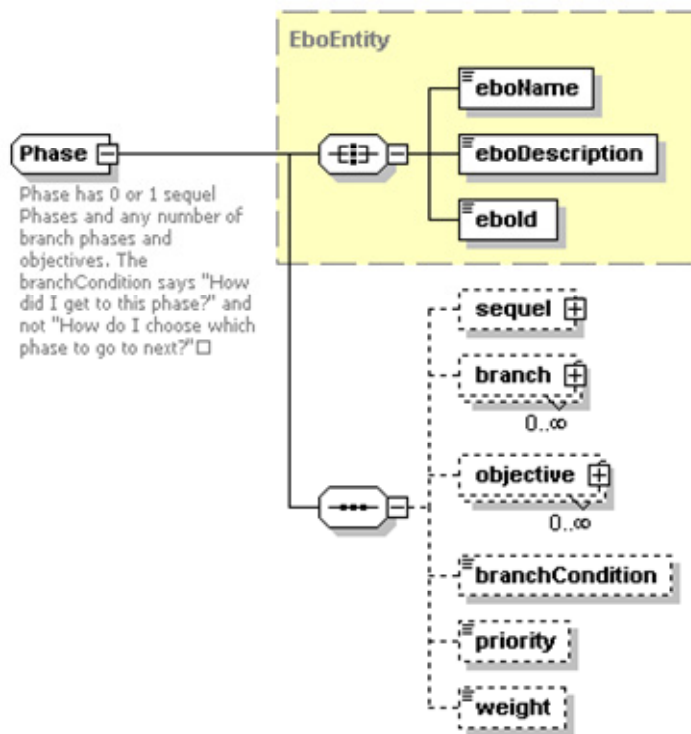
4.3.19.   PlanningDuration

diagram

type   extension of **EboEntity**

used by   elements   **Effect/desiredDuration Task/duration**

annotation   documentation   Used to represent a duration as value and unitOfTime.

source
```xsd
<xsd:complexType name="PlanningDuration">
 <xsd:annotation>
  <xsd:documentation>Used to represent a duration as value and unitOfTime.</xsd:documentation>
 </xsd:annotation>
 <xsd:complexContent>
  <xsd:extension base="EboEntity">
   <xsd:all>
    <xsd:element name="value" type="xsd:string" minOccurs="0"/>
    <xsd:element name="unitOfTime" type="sdt:TimeUnit" minOccurs="0"/>
   </xsd:all>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>
```

4.3.20. ReferenceDateDefinition

diagram

type   extension of **EboEntity**

used by   element   **Plan/referenceDateDefinition**

annotation   documentation   This is used to represent either Calendar or ReferenceDates. With this, we can say that C + some calendar date and D + 0 maps to C + 5 and so on. The type is the letter associated wi ReferenceDateDefinition- C,D,etc. The origin is either a Calendar or a Reference date.The origin was added for better SMS integration. This represents the number of milliseconds fro beginning of the epoch until the origin.

source

```xml
<xsd:complexType name="ReferenceDateDefinition">
 <xsd:annotation>
   <xsd:documentation>This is used to represent either Calendar or ReferenceDates.  With this, we can say
that C + 0 maps to some calendar date and D + 0 maps to C + 5 and so on. The type is the letter associated
with this ReferenceDateDefinition- C,D,etc. The origin is either a Calendar or a Reference date.The millisecond
origin was added for better SMS integration.  This represents the number of milliseconds from the beginning of
the epoch until the origin.</xsd:documentation>
 </xsd:annotation>
 <xsd:complexContent>
  <xsd:extension base="EboEntity">
   <xsd:all>
    <xsd:element name="type" type="xsd:string"/>
    <xsd:element name="timeUnits" type="sdt:ReferenceDateTimeUnitType"/>
    <xsd:element name="origin" type="CalendarOrReferenceDate"/>
    <xsd:element name="millisecondOrigin" type="xsd:long"/>
    <!-- type = C, D, etc -->
    <!--  origin is either reference or a calendar date. -->
    <!-- added in SDT 1.5.3 -->
   </xsd:all>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>
```
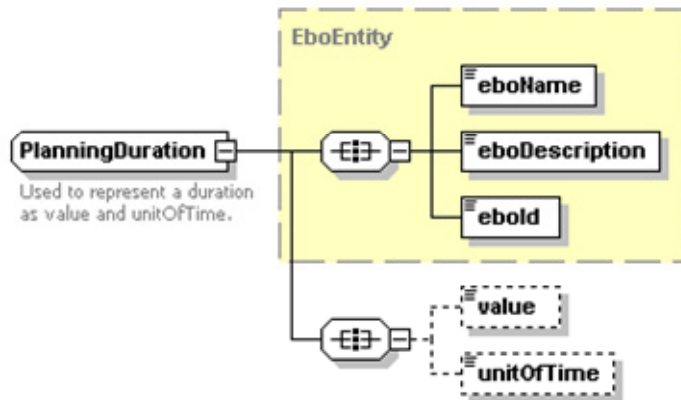
4.3.21.   RestrictionConstraint

diagram

type   extension of **EboEntity**

used by   element   **EffectRestrictionNode/constraint**

annotation documentation   RestrictionConstraints are comprised of 4 plain text fields. No structure is provided here to n
                            flexibility and interoperability with other tools.

source   ```xsd
<xsd:complexType name="RestrictionConstraint">
  <xsd:annotation>
    <xsd:documentation>RestrictionConstraints are comprised of 4 plain text fields. No structure is provided
here to maximize flexibility and interoperability with other tools.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:sequence>
        <xsd:element name="targetedElement" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="comparator" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="valueCompared" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="units" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
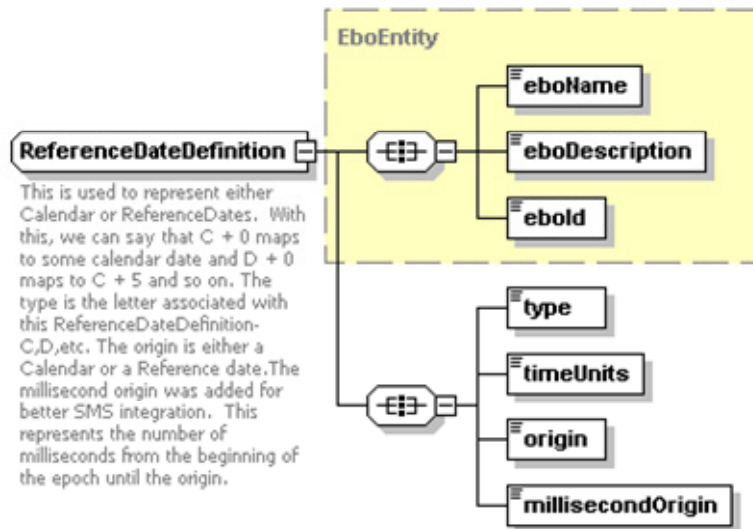   </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

## 4.3.22.　　RulesOfEngagement

diagram



| | |
|---|---|
| type | extension of **EboEntity** |
| used by | element **MissionAnalysis/rulesOfEngagement** |
| annotation | documentation RulesOfEngagement contains constraints and restraints.  Constraint represents something you must do. Restraint represents something you cannot do. |
| source | |

```
<xsd:complexType name="RulesOfEngagement">
 <xsd:annotation>
  <xsd:documentation>RulesOfEngagement contains constraints and restraints.  Constraint represents
something you must do. Restraint represents something you cannot do.</xsd:documentation>
 </xsd:annotation>
 <xsd:complexContent>
  <xsd:extension base="EboEntity">
   <xsd:sequence>
    <xsd:element name="constraint" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="restraint" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
   </xsd:sequence>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>
```

4.3.23. SituationElement

diagram

type	extension of **EboEntity**
used by	elements **Effect/indirectObjectOfEffect Task/indirectObjectOfTask Effect/locationOfEffect Task/locationOfTask Standard/measuredObject Effect/objectOfEffect Task/objectOfTas**
	complexType **Target**

source
```xsd
<xsd:complexType name="SituationElement">
  <xsd:complexContent>
    <xsd:extension base="EboEntity">
      <xsd:all>
        <xsd:element name="type" type="xsd:string" minOccurs="0"/>
        <xsd:element name="instanceID" type="xsd:string" minOccurs="0"/>
        <xsd:element name="annotation" type="xsd:string" minOccurs="0"/>
      </xsd:all>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

4.3.24.   Standard
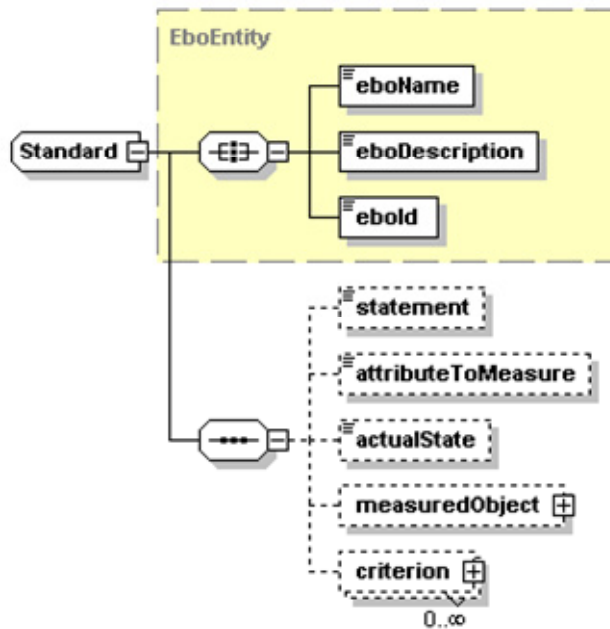
type   extension of **EboEntity**

used by   elements   **Effect/desiredEffect Effect/indicator CausalLinkage/indicator Task/indicator
                           Effect/measureOfEffectiveness Task/measureOfEffectiveness**

source   <xsd:complexType name="Standard">
              <xsd:complexContent>
                <xsd:extension base="EboEntity">
                  <xsd:sequence>
                    <xsd:element name="statement" type="xsd:string" minOccurs="0"/>
                    <xsd:element name="attributeToMeasure" type="xsd:string" minOccurs="0"/>
                    <xsd:element name="actualState" type="xsd:string" minOccurs="0"/>
                    <xsd:element name="measuredObject" type="sdt:SituationElement" minOccurs="0"/>
                    <xsd:element name="criterion" type="sdt:Criterion" minOccurs="0" maxOccurs="unbounded"/>
                  </xsd:sequence>
                </xsd:extension>
              </xsd:complexContent>
              </xsd:complexType>

171

4.3.25. Target
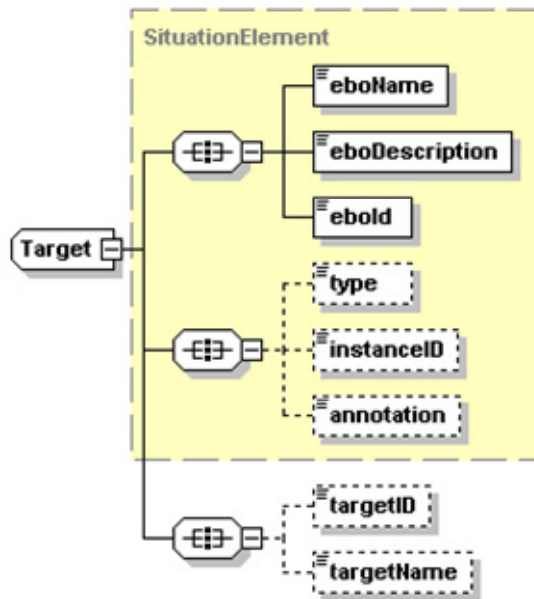
type   extension of **SituationElement**

used by   elements   **Effect/target Task/target**
           complexType   **MIDBTarget**

source   <xsd:complexType name="Target">
           <xsd:complexContent>
            <xsd:extension base="SituationElement">
             <xsd:all>
              <xsd:element name="targetID" type="xsd:string" minOccurs="0"/>
              <xsd:element name="targetName" type="xsd:string" minOccurs="0"/>
             </xsd:all>
            </xsd:extension>
           </xsd:complexContent>
          </xsd:complexType>

4.3.26. **Task**
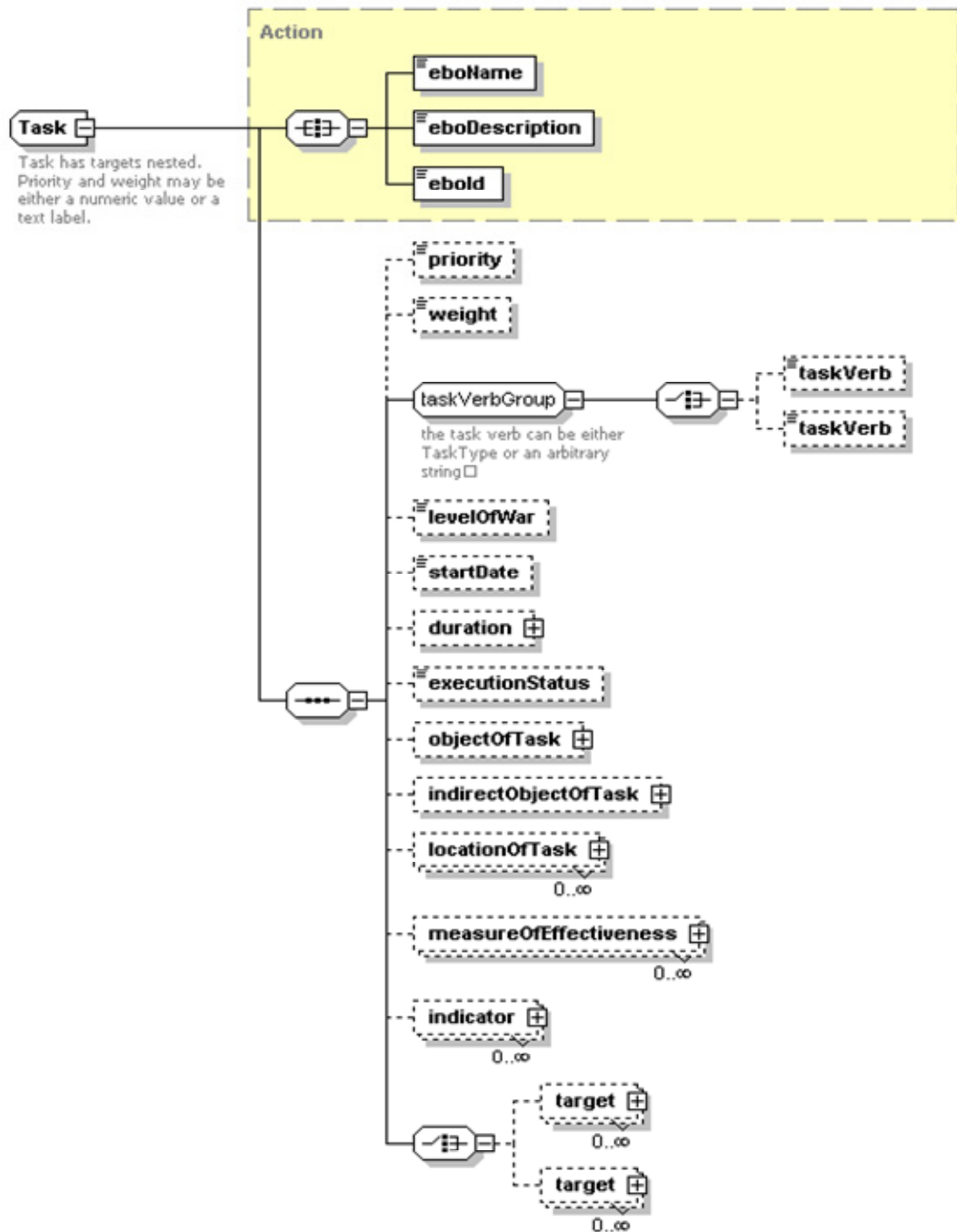
type	extension of **Action**
used by	element **CausalLinkage/taskTrigger**
annotation	documentation Task has targets nested. Priority and weight may be either a numeric value or a text label.
source	`<xsd:complexType name="Task">`
	`<xsd:annotation>`
	`<xsd:documentation>`Task has targets nested. Priority and weight may be either a numeric value or a text

4.3.27.　TemporalConstraint
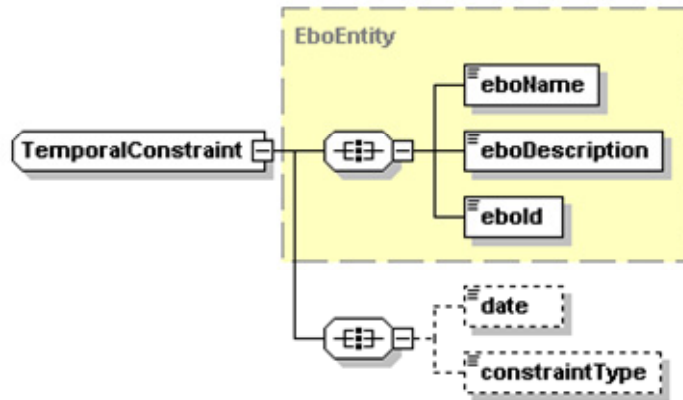
extension of **EboEntity**

elements **Criterion/constraintType Criterion/temporalConstraint**

```
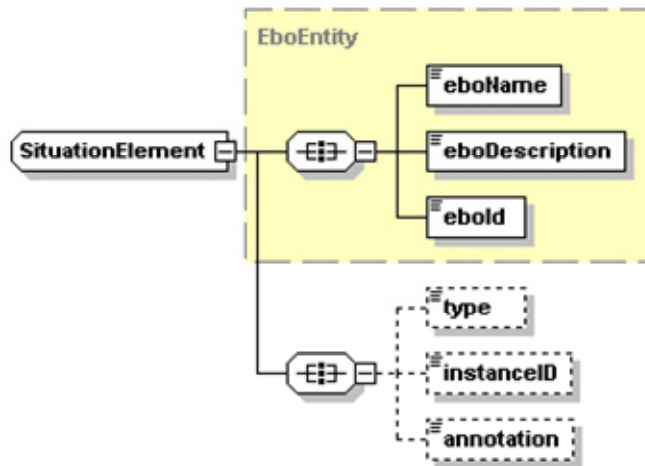<xsd:complexType name="TemporalConstraint">
 <xsd:complexContent>
  <xsd:extension base="EboEntity">
   <xsd:all>
    <xsd:element name="date" type="sdt:CalendarOrReferenceDate" minOccurs="0"/>
    <xsd:element name="constraintType" type="sdt:TemporalConstraintType" minOccurs="0"/>
   </xsd:all>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>
```

4.3.28. UITargetNode

diagram

type	extension of **MIDBTarget**	
used by	elements	**DNSListNode/ueTarget** **EffectRestrictionNode/ueTarget**
annotation	documentation	UITargetNodes (Unintended target nodes) extends MIDBTargets, but do not currently provide additional information.

source

```xsd
<xsd:complexType name="UITargetNode">
  <xsd:annotation>
    <xsd:documentation>UITargetNodes (Unintended target nodes) extends MIDBTargets, but do not currently
provide any additional information.</xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="MIDBTarget"/>
  </xsd:complexContent>
</xsd:complexType>
```

4.4. SIMPLE TYPES

4.4.1. CalendarDate

type	restriction of **xsd:string**	
used by	simpleType	**CalendarOrReferenceDate**
facets	pattern	[0-31][ ][Jan|Feb|Mar|Apr|May|Jun|Jul|Aug|Sep|Oct|Nov|Dec][ ][2000-2020][ ] [0-23][:][0-59]
annotation	documentation	calendar dates are as follows: DD Mon YYYY HH:MM Example: 14 Oct 2003 13:00

source
```xml
<xsd:simpleType name="CalendarDate">
  <xsd:annotation>
    <xsd:documentation>calendar dates are as follows: DD Mon YYYY HH:MM Example:  14 Oct 2003
13:00</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-31][ ][Jan|Feb|Mar|Apr|May|Jun|Jul|Aug|Sep|Oct|Nov|Dec][ ][2000-2020][ ] [0-
23][:][0-59]"/>
  </xsd:restriction>
</xsd:simpleType>
```

4.4.2. CalendarOrReferenceDate

type	union of (**CalendarDate**, **ReferenceDate**)	
used by	elements	**Criterion/date TemporalConstraint/date Effect/desiredStartDate ReferenceDateDefinition/origin Task/startDate**
annotation	documentation	Union of Calendar and Reference Date types

source
```xml
<xsd:simpleType name="CalendarOrReferenceDate">
  <xsd:annotation>
    <xsd:documentation>Union of Calendar and Reference Date types</xsd:documentation>
  </xsd:annotation>
  <xsd:union memberTypes="CalendarDate ReferenceDate"/>
</xsd:simpleType>
```

4.4.3. EffectType

type	restriction of **xsd:string**	
used by	element	**effectVerbGroup/effectVerb**
facets	enumeration	annihilate
	enumeration	attrit
	enumeration	coerce
	enumeration	deny
	enumeration	disrupt
	enumeration	paralyze
	enumeration	decapitate
	enumeration	deceive
	enumeration	defend
	enumeration	degrade
	enumeration	delay
	enumeration	destroy
	enumeration	deter
	enumeration	disable
	enumeration	dislocate
	enumeration	divert
	enumeration	drive
	enumeration	expel
	enumeration	isolate
	enumeration	sever
	enumeration	stop
annotation	documentation	These are effect types currently supported in SDT, but users may type in other effect verbs

source

```xml
<xsd:simpleType name="EffectType">
 <xsd:annotation>
  <xsd:documentation>These are effect types currently supported in SDT, but users may type in other effect
verbs as desired.</xsd:documentation>
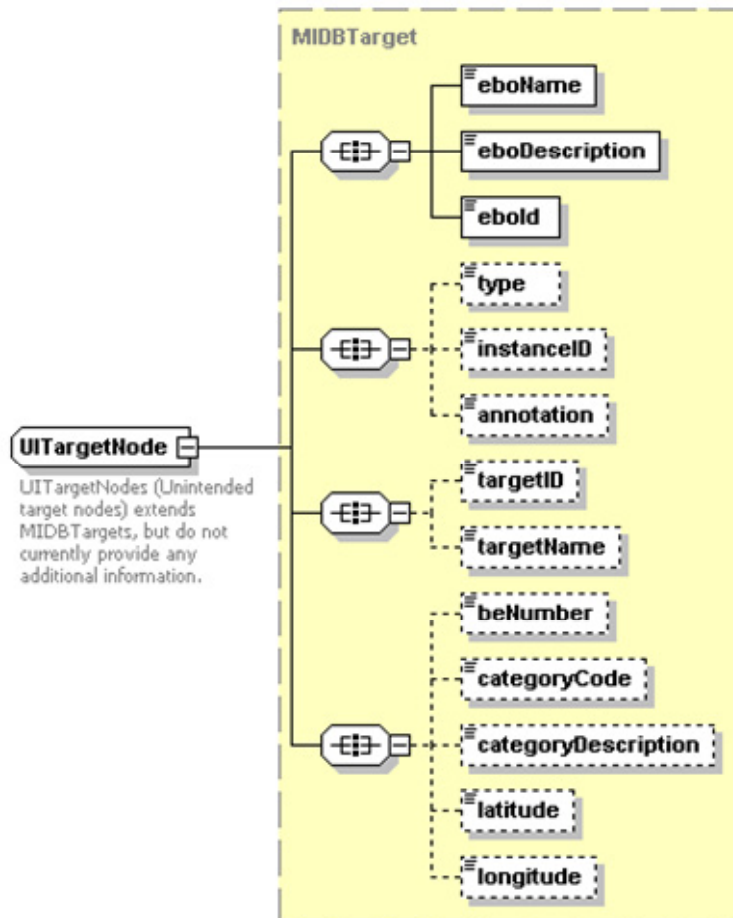 </xsd:annotation>
 <xsd:restriction base="xsd:string">
  <xsd:enumeration value="annihilate"/>
  <xsd:enumeration value="attrit"/>
  <xsd:enumeration value="coerce"/>
  <xsd:enumeration value="deny"/>
  <xsd:enumeration value="disrupt"/>
  <xsd:enumeration value="paralyze"/>
  <xsd:enumeration value="decapitate"/>
  <xsd:enumeration value="deceive"/>
  <xsd:enumeration value="defend"/>
  <xsd:enumeration value="degrade"/>
  <xsd:enumeration value="delay"/>
  <xsd:enumeration value="destroy"/>
  <xsd:enumeration value="deter"/>
  <xsd:enumeration value="disable"/>
  <xsd:enumeration value="dislocate"/>
  <xsd:enumeration value="divert"/>
  <xsd:enumeration value="drive"/>
  <xsd:enumeration value="expel"/>
  <xsd:enumeration value="isolate"/>
  <xsd:enumeration value="sever"/>
  <xsd:enumeration value="stop"/>
 </xsd:restriction>
</xsd:simpleType>
```

4.4.4. LevelOfWar

type	restriction of **xsd:string**	
used by	elements	**Objective/levelOfWar** **Task/levelOfWar**
facets	enumeration	Operational
	enumeration	Tactical
	enumeration	National
	enumeration	Strategic
source		

```xml
<xsd:simpleType name="LevelOfWar">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="Operational"/>
    <xsd:enumeration value="Tactical"/>
    <xsd:enumeration value="National"/>
    <xsd:enumeration value="Strategic"/>
  </xsd:restriction>
</xsd:simpleType>
```

4.4.5. PointOfViewType

type	restriction of **xsd:string**	
used by	element	**Plan/pointOfView**
facets	enumeration	blue
	enumeration	red
	enumeration	gray
annotation	documentation	A plan may be authored from the blue (friendly), red (enemy) or gray (neutral) perspective
source		

```xml
<xsd:simpleType name="PointOfViewType">
  <xsd:annotation>
    <xsd:documentation>A plan may be authored from the blue (friendly), red (enemy) or gray (neutral)
perspective</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="blue"/>
    <xsd:enumeration value="red"/>
    <xsd:enumeration value="gray"/>
  </xsd:restriction>
</xsd:simpleType>
```

4.4.6. ReferenceDate

type	restriction of **xsd:string**	
used by	simpleType	**CalendarOrReferenceDate**
facets	pattern	[C-Z|c-z][+-][0-9]+
annotation	documentation	Reference days are represented as follows: [Type][+/-][Val] Example: C+3, D-5, L+360, H-24
source		

```xml
<xsd:simpleType name="ReferenceDate">
  <xsd:annotation>
    <xsd:documentation>Reference days are represented as follows: [Type][+/-][Val] Example: C+3, D-5,
L+360, H-240                              </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[C-Z|c-z][+-][0-9]+"/>
  </xsd:restriction>
</xsd:simpleType>
```

4.4.7. ReferenceDateTimeUnitType

type	restriction of **xsd:string**	
used by	element	**ReferenceDateDefinition/timeUnits**
facets	enumeration	hours
	enumeration	days
annotation	documentation	the time units for reference dates must be either hours or days

source
```
<xsd:simpleType name="ReferenceDateTimeUnitType">
  <xsd:annotation>
    <xsd:documentation> the time units for reference dates must be either hours or days
          </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="hours"/>
    <xsd:enumeration value="days"/>
  </xsd:restriction>
</xsd:simpleType>
```

4.4.8. TaskType

type	restriction of **xsd:string**
used by	element **taskVerbGroup/taskVerb**
facets	enumeration block
	enumeration destroy
	enumeration deter
	enumeration disrupt
	enumeration degrade
	enumeration decapitate
	enumeration divert
	enumeration dislocate
	enumeration delay
	enumeration deny
	enumeration deceive
	enumeration defend
	enumeration interdict
	enumeration drive
	enumeration expel
	enumeration isolate
	enumeration sever
	enumeration stop
annotation	documentation These are Task types currently supported in SDT, but users may type in other task verbs as

source

```xml
<xsd:simpleType name="TaskType">
  <xsd:annotation>
    <xsd:documentation>These are Task types currently supported in SDT, but users may type in other task
verbs as desired.</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="block"/>
    <xsd:enumeration value="destroy"/>
    <xsd:enumeration value="deter"/>
    <xsd:enumeration value="disrupt"/>
    <xsd:enumeration value="degrade"/>
    <xsd:enumeration value="decapitate"/>
    <xsd:enumeration value="divert"/>
    <xsd:enumeration value="dislocate"/>
    <xsd:enumeration value="delay"/>
    <xsd:enumeration value="deny"/>
    <xsd:enumeration value="deceive"/>
    <xsd:enumeration value="defend"/>
    <xsd:enumeration value="interdict"/>
    <xsd:enumeration value="drive"/>
    <xsd:enumeration value="expel"/>
    <xsd:enumeration value="isolate"/>
    <xsd:enumeration value="sever"/>
    <xsd:enumeration value="stop"/>
  </xsd:restriction>
</xsd:simpleType>
```

4.4.9. TemporalConstraintType

type	restriction of **xsd:string**
used by	element **TemporalConstraint/constraintType**
facets	enumeration achieve-by enumeration maintain-until

source	`<xsd:simpleType name="TemporalConstraintType">`
	  `<xsd:restriction base="xsd:string">`
	   `<xsd:enumeration value="achieve-by"/>`
	   `<xsd:enumeration value="maintain-until"/>`
	  `</xsd:restriction>`
	`</xsd:simpleType>`

4.4.10.   TimeUnit

type	restriction of **xsd:string**
used by	element **PlanningDuration/unitOfTime**
facets	enumeration years enumeration months enumeration weeks enumeration days enumeration hours enumeration minutes enumeration seconds

source	`<xsd:simpleType name="TimeUnit">`
	  `<xsd:restriction base="xsd:string">`
	   `<xsd:enumeration value="years"/>`
	   `<xsd:enumeration value="months"/>`
	   `<xsd:enumeration value="weeks"/>`
	   `<xsd:enumeration value="days"/>`
	   `<xsd:enumeration value="hours"/>`
	   `<xsd:enumeration value="minutes"/>`
	   `<xsd:enumeration value="seconds"/>`
	  `</xsd:restriction>`
	`</xsd:simpleType>`

5. APPENDICES

5.1. ACRONYM LIST

API	Application Program Interface. Defines the interface from an external application.
CAT	Causal Analysis Tool
COA	Course of Action
COG	Center of Gravity
EBO	Effects Based Operations
SOAP	Simple Object Access Protocol
SDP	Software Development Plan
SDT	Strategy Development Tool
SMS	Strategy Management Service
SRS	Software Requirements Specification. Allocates software requirements to CSCIs.
WSDL	Web Services Definition Language
XML	eXtensible Markup Language
XSD	XML Schema Definition
XSL	XML Stylesheet Language

5.2. DIAGRAM KEY
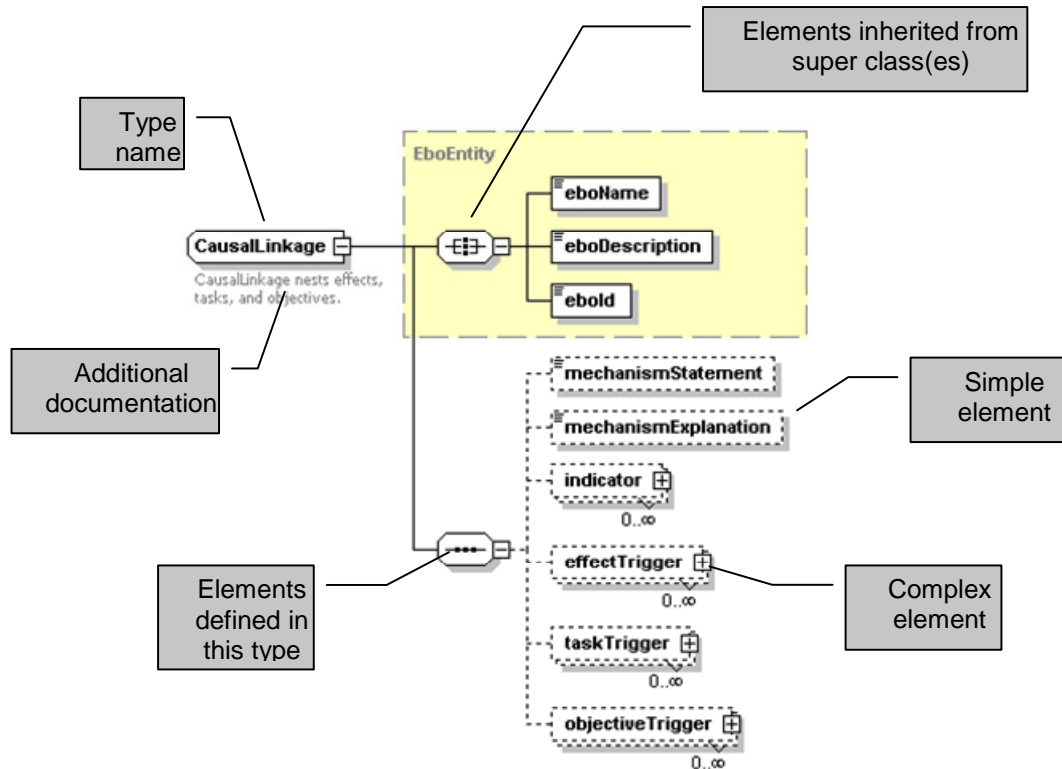
<element | group | complexType | simpleType> <**typeName**>

type	The name of the super class (if any)
used by	List of parent elements that make use of this type
facets	Any restrictions on the element's contents, such as an enumeration or a pattern.
source	A detailed description of the individual elements using the XSD syntax.

Figure 56: Sample plan element

APPENDIX D

JEFX '04 EBO TECHNICAL ASSESSMENT RESULTS

EBO

The Strategy Development Tool (SDT) provides a suite of plan authoring and Course of Action (COA) analysis tools for joint air campaign planning. The SDT primarily supports Mission Analysis and COA Development, the first two steps in the Joint Air Estimate Process (JAEP).

The SDT's plan authoring tools use an effects-based plan representation for decomposing the Commander's Guidance into COAs, phases, objectives, effects, causal linkages, tasks, and targets. The SDT is integrated with the following effects-based analysis tools to support the third step of the Joint Air Estimate Process (JAEP), COA Analysis:

- Blue COAs may be analyzed within AFRL's Operational Assessment Tool (OAT) to determine the probability of achieving the objectives in the COA over time.

- Adversary center of gravity (COG) models with blue interventions may be analyzed in OAT to assess the impact of interventions on enemy systems.

- Target Systems Analysis may be used for target system visualization and analysis to predict the enemy's response to outages induced by strike targets in the COA.

- Option Generation may be used to generate strike target sets to achieve desired effects related to certain target systems on a specified set of affected facilities.

During Spiral 3 and MainEx, the SDT was successfully utilized to build, save, edit, and load stored COG models. The building, saving, editing, and loading of stored COAs was also successfully demonstrated. COAs were created with sequel and branch phases using the graphical phase editor and objectives were edited using the objectives editor. During COA development, the capability to incorporate objectives, desired effects, tasks and targets, and causal links into COA design was also successfully verified.

The SDT COG Articulator also enabled development of COG models from scratch and via loading of effects-based mission/strategy templates. After development of causal adversarial models, the SDT COG Articulator was utilized to select a causal chain, which was later exported into the effects based plan editor. Target sets were also associated with tasks and effects for incorporation into the effects-based plan. Measures of Effectiveness (MOEs)/Indicators were specified for effects, tasks, and mechanisms (causal links) in the COA and used to identify potential COA options.

Commander's Intent and Mission Analysis data was successfully imported from TBMCS/Strategy Management Service (SMS). Data field inconsistencies exhibited during Spiral 2 relating to the export and import of COAs/Mission Analysis data to and from the Strategy Planning Tool (SPT) via SMS were resolved for (except for SDT/SPT field implementation differences). During Spiral 3 and MainEx, importing and exporting of extensive COAs (8AF

generated for JEFX04) and Mission Analysis Data between SPT and SDT via SMS were successful.

The SDT Target Set Tool successfully provided the means to specify no strike targets and do not affect targets, based on the specified ROE. The Target Set Tool also provided the capability to perform target set analysis and visualization of links and nodes within multiple target systems. Created plan effects and selected multiple affected targets based on a query. Resultant targets were successfully sent from Target Tool Set to the SDT Plan Editor. Target sets (Affected Targets, Direct Targets, Do Not Strike, and Do Not Affect Targets) were then associated with tasks and effects for incorporation into the effects-based plan. SDT Option Generation automatically provided target set options for achieving disruption effects based on Electric Power (EP) and Petroleum Oil Lubricants (POL) target system information. For JEFX 04, MIDB link data was only provided within like-target systems, not across differing target systems (i.e. disruption of EP nodes could only be accomplished by affecting other EP nodes).

Within SDT, the Timeline Editor was successfully utilized to view and edit temporal information for effects-based plan elements. As Start, Duration and End times were modified; changes to associated temporal information were appropriately reflected in the Parent-Child rollups.

COA Analysis was also conducted to analyze the developed models using the OAT, and blue interventions against red COG nodes were analyzed. During MainEx, models were comprised of over 300 nodes, with over 500 blue causal links. Blue intervention causal strength was adjusted and the probability profile was updated accordingly. Evidence was also provided from PBA/AF2TK via email, and manually entered to assess revised probabilities. COAs with multiple varied interventions were analyzed and it was verified that the probability profile for each critical node was impacted. Additional intervention events were scheduled within a given period and the probability profile again reflected the additional intervention events. Although probability profiles reflected impacts of blue interventions and causal parameter modifications, 8AF noted potential deficiencies in the probability profile algorithms with respect to reconstitution time modeling. The initiative provider agreed that refinement of the algorithms was necessary. Recommend accuracy of probability profile algorithms be validated during formal testing.

The capability to import A2IPB products (NAIs, TAIs and Red COAs) for mission analysis visualization was successfully demonstrated. A2IPB products were sent via email in XML format. Once the email was received, the operator selected the XML and saved it in the directory. A2IPB products were then successfully displayed on the map and in table format.

The interface with Information Warfare Planning Capability (IWPC) was also successfully utilized. COAs were exported to SMS and imported by IWPC. In regards to the capability to export Candidate Target Lists (CTLs) to the Joint Targeting Toolbox (JTT), the provider did not receive the JTT schema in sufficient time to integrate this functionality.

In regards to SDT's capability to export MOE information for use by ISR collection managers, SDT currently supports exporting indicators in XML. However, in the absence of a machine-to-machine interface via SMS, a manual process was successfully used to input the information into the PBA toolset (AF2TK) during MainEx.

For MainEx, the initiative provider integrated new 8th AF-requested capabilities for effects-based plan collaboration and COA comparison/assessment. During MainEx, these capabilities were technically assessed with the following results:

Collaboration – During branch COA planning, the collaboration capability provided multiple users with the ability to successfully edit and collaborate on the same plan from different workstations. As updates were processed, they were displayed for other users in real-time. In a collaborative environment, a new plan was uploaded to the database on the collaboration server. Once the upload was complete, multiple users were able to open a collaborative plan browser, select the uploaded plan, and download the plan to SDT. During Mission Analysis, users were able to lock and edit data panes to prevent concurrent editing of the same data. As Mission Analysis data was locked for a particular user, indicators appeared on other users' workstations as notification that another user had locked the data for editing. When edits were complete and unlocked, other users were able to view the updated information. Nodes were also updated by locking out other users during the change and then presenting the new data once released by the original editor.

Existing file based plans were also successfully uploaded to the collaboration server and downloaded by different users for collaboration.

COA Comparison – Prior to MainEx, the COA comparison capability was integrated into SDT. This provided the operator with the capability to enter numeric scores based on operator-tailored criteria in a COA Comparison Matrix window. Scores were entered for multiple phases for the following criteria: Suitability, Feasibility, Acceptability, Completeness and Probability of Success. The Matrix was then successfully exported to MS Excel for operator comparison with other COAs.

Final Results:

Total Core Capabilities: 43

Successful: 40

Partially Successful: 2

Not Demonstrated: 1

Number	Initiative
04-027	Predictive Battlespace Awareness /Effects Based Operations (PBA/EBO)

#	Capability	MOP/MOE/THRESHOLD	Location	Eval	Comments
EBO_1.01	SDT - Provides a capability to build, save, edit, and load stored Center of Gravity (COG) models.	Create and save COG models. Verify stored COG models can be retrieved, edited and saved.		S	Sp1: Created and saved COG model from scratch. Reloaded the saved COG model. Also, loaded and edited pre-stored model. Sp2 - reassessed Sp3 - Not reassessed MainEx - Not reassessed, but successfully utilized operationally
EBO_1.02	SDT - Provides a capability to build, save, edit, and load stored Courses of Action (COAs) with phases, objectives, desired effects, unintended effects, tasks, targets, and do-not-strke/affect targets.	Create and save COAs with phases, objectives, desired effects, and tasks. Verify stored COAs can be retrieved, edited and saved. Also, verify causal links and targets can be assigned to each task.including specification of a plan name, description, point of view, facts, assumptions, and reference times.		S	Sp1: Created and saved COA with phases (incl sequel and branch phases). Retrieved and edited COA. Stored, reloaded and verify changes. Sp2 - reassessed Sp3 - Not reassessed MainEx - Not reassessed, but successfully utilized operationally
EBO_1.03	SDT - Provides a capability to specify a plan with 1 or more COAs and mark one of these COAs as the selected COA.	Demonstrate that a plan with a one or more COAs can be created. A plan can be specified with a plan name, description, point of view, facts, assumptions, reference times, and which COA has been selected for further plan elaboration.		S	Sp3 - Selecting a COA newer than COA 1 (COA 2 or greater)and saving in Relations tab results in the loss of earlier COAs (for further plan elaboration). This problem has a workaround - Save COAs incrementally. Fix planned for MainEx. MainEx - Random COAs were successfully marked as the selected COA
EBO_1.04	SDT - Provides a capability to specify phases for a COA. A phase can be specified as being an initial, sequel, or branch phase. A branch condition can be specified for branch phases.	Demonstrate that phases can be created for a COA. Verify that a phase type and branch condition (for phases of type branch) can be assigned to each phase.		S	Sp3 - phases were created and phase types and branch conditions were assigned to each phase MainEx - Not reassessed, but successfully utilized operationally
EBO_1.05	SDT - Provides a capability to specify strategic, operational, and tactical level objectives for a COA.	Demonstrate that an objective can be created with an assigned planning level.		S	Sp3 - objectives were successfully created with assigned planning levels (strategic, operational, and tactical) MainEx - Not reassessed, but successfully utilized operationally
EBO_1.06	SDT - Provides a capability to specify desired and unintended effects and subeffects for objectives. An effect can be specified with timing and location information.	Demonstrate that an effect can be created. Verify a effect verb, COG type, object of effect, indirect object of effect, planning level, desired effect level, desired start date and duration of effect, delay and persistence of effect, and locations of effects can be specified for each effect.		S	Sp3 - desired effects, unintended effects and subeffects for objectives were successfully specified with all attributes MainEx - Not reassessed, but successfully utilized operationally
EBO_1.07	SDT - Provides a capability to specify causal links for effects. A causal link can be specified with a causal strength, delay, and persistence.	Demonstrate that a causal link can be created.. Verify a causal strength, delay, persistence, and indicators can be specified for each causal link.		S	Sp3 - causal links were successfully created with defined causal strength, delay, persistence, and indicators MainEx - Not reassessed, but successfully utilized operationally
EBO_1.08	SDT - Provides a capability to specify tasks associated with causal links. A task can be specified with a probability of success, planning level, timing, and location information.	Demonstrate that a task can be created. Verify task verb, object of task, probability of success, planning level, desired start date and duration of the task, and locatons of tasks can be specified for each task.		S	Sp3 - tasks were associated with causal links (including all attributes) MainEx - Not reassessed, but successfully utilized operationally
EBO_1.09	SDT - Provides a capability to specify targets associated with effects, targets, do not strike lists, and effect restriction lists. A target can be specified with a target type, BE number, target category code, and geospacial location (lat/lon) information.	Demonstrate that a target can be created. Verify a target type, BE number, category code, and geospacial location (lat/lon) can be specified for each target.		S	Sp3 - Targets were created target can be created with the following attributes: target type, BE number, category code, and geospacial location (lat/long) MainEx - Not reassessed, but successfully utilized operationally
EBO_1.10	SDT - Provides a capability to specify priority and weight of effort for phases, objectives, desired effects, and tasks.	Verify priority and weight of effort can be specified for phases, objectives, desired effects, and tasks.		S	Sp3 - priority and weight of effort were specified for phases, objectives, desired effects, and tasks. MainEx - Not reassessed, but successfully utilized operationally

#	Capability	MOP/MOE/THRESHOLD	Location	Eval	Comments
EBO_1.11	SDT - Provides the capability to graphically edit COAs.	Verify phases, objectives, effects, causal links, tasks, and targets can be created and edited graphically in the Structural Editor.		S	Sp1: Created COA phases and sequels/branches using graphical phase editor. Verified objectives could be edited using the objectives editor. MainEx - Not reassessed, but successfully utilized operationally Sp2 - reassessed Sp3 - not reassessed
EBO_1.12	SDT/OAT - Provides a capability to perform red COG analysis.	Demonstrate the ability to identify critical nodes of vulnerability in enemy COGs through probabilistic analysis of interventions.		P	Sp1: Performed analysis by using COAs with multiple varied interventions and ensured probability profile (probability over time) for each critical node was impacted. Scheduled additional intervention events within a given period and ensured probability profile reflected additional intervention events. Removal of the intervention is also reflected. During Sp1, unable to perform analysis of large COG models with latest CAT version. Had to load previous version for analysis. However, successful during Sp2. Sp3 - reassessed successfully MainEx - Although probability profiles reflected impacts of blue interventions and causal parameter modifications, 8AF noted potential deficiencies in the probability profile algorithms with respect to reconstitution time modeling. The initiative provider agreed that refinement of the algorithms was necessary. Recommend accuracy of probability profile algorithms be validated during formal testing
EBO_1.13	SDT - Provides a capability to select a causal chain and export it into the effects based plan.	Demonstrate the ability to select a causal chain and export it to the effects based plan. Demonstrate the ability to display an imported causal chain in the effects based plan.		S	Sp2 - Using the COG Articulator, selected causal chain and exported it to SDT. Plan fragments were added to the plan. Sp3 - Reassessed successfully
EBO_1.14	SDT/OAT - Provides the capability to analyze Blue COAs	Demonstrate the ability to export a blue COA to OAT. Demonstrate the ability to analyze the COA in OAT.		S	
EBO_1.15	SDT - Target Set Tool provides a capability to visualization of links and nodes within multiple target systems.	Verify links and nodes are displayed on the Target Set Tool map.		S	Sp1: Links and nodes are present on the map. Selected target sets and classified them. Verified selected targets were mapped. Also verified targets could be automatically proposed an manually refined prior to incorporation into the effects-based plan. Did not have CPLEX license for the MATLAB application for predictive analysis of effects on enemy target systems. Sp2 - reassessed. Anomaly noted with targets listed more than once in target set - provider suspects potential MIDB schema issue. Sp3 - Created plan effects and selected multiple affected targets based on a query. Resultant targets were sent from Target Tool Set to the SDT Plan Editor successfully Sp3 - links and nodes were properly displayed on the Target Set Tool map MainEx - Successfully utilized operationally
EBO_1.17	SDT - Provides a capability to automatically provide target options for achieving disruption effects based on EP and POL target system information.	Verfiy option generation proposes target set options when EP and POL target set information is available.		S	Sp3 - option generation successfully proposed target set options for Electric Power (EP) and Petrolium Oil (POL) facilities
EBO_1.18	SDT - Provides a capability to specify target sets for association with tasks and effects for incorporation into the effects-based plan.	Demonstrate the ability to specify targets as Affected Targets, Direct Targets, Do Not Strike, and Do not Affect Target and export target sets to the effects-based plan. Verify selected target sets are exported to the effects-based plan.		S	Sp1: Verified selected targets were incorporated into the effects-based plan. Sp2 - reassessed Sp3 - reassessed successfully MainEx - Successfully utilized operationally

#	Capability	MOP/MOE/THRESHOLD	Location	Eval	Comments
EBO_1.19	SDT - Provides a capability to specify MOEs/Indicators for objectives, effects, tasks, and causal links in the COA.	Demonstrate the ability to specify MOEs/Indicators for objectives, effects, tasks, and causal links in the COA.		S	Sp1: Specified MOEs and indicators for effects, tasks. Causal links do not have MOEs (by design) - specified indicators. Sp2 - reassessed Sp3 - not reassessed MainEx - Successfully utilized operationally
EBO_1.20	SDT - Provides a capability to specify commander`s intent.	Demonstrate the ability to specify commander`s intent.		S	Sp1: Commander's intent may be exportable to SMS in later spirals - requires SMS client interface. Sp2 - reassessed. Sp3 - not reasssessed MainEx - Successfully utilized operationally
EBO_1.21	SDT - Provides a capability to specify mission analysis data	Demonstrate the ability to specify mission analysis data.		S	Sp1: Currently done via manual entry. TBD - Automatic entry may be via import from TBMCS/SMS in later spirals. Mission analysis data may also be exportable to SMS. Sp2 - reassessed Sp3 - not reassessed MainEx - Successfully utilized operationally
EBO_1.22	SDT - Provides a capability to specify no strike targets and do not affect targets	Demonstrate the ability to specify no strike and do not affect targets. Verfiy Target Set Tool displays no strike and do not affect targets and Plan editor reflects no strike and do not affect targets.		S	Sp1: Chose undesired effects and selected Hospitals and POL facilities as do not affect within the plan editor. Selected Hoover Dam as no strike via Target set tool. Also selected specific POL facilities using the Query Tool. Sp2 - reassessed. With Do Not Affect Targets, user can specify restrictions. Sp3 - not reassessed MainEx - Successfully utilized operationally
EBO_1.23	SDT - Provides a capability to view and edit timing information for an effects-based plan.	Demonstrate the ability to view and edit temporal information for plan elements in the Timeline Editor.		S	Sp3 - For Parent-Child rollups, modified the Start, Duration and End times. Effects are rolled up in the Timeline Editor, but Tasks are not rolled up. MainEx - Timeline Editor was successfully utilized to view and edit temporal information for effects-based plan elements. As Start, Duration and End times were modified; changes to associated temporal information were appropriately reflected in the Parent-Child rollups
EBO_1.24	SDT - Provides an Effects Dictionary via integrated help functionality	Verify the ability to look up definitions and example usages of effect verbs		S	Sp2 - reassessed Sp3 - not reassessed Sp3 - not reassessed
EBO_1.25	SDT - provides the capability to build, save and load mission/strategy templates	Create and save multiple plan fragments (packages). Create new plan and insert plan fragments.		S	Sp1: Inserted previously built fragments into an existing plan. Verified fragment was incorporated. Sp2 - reassessed. Saved fragments from plan and loaded into new plan. Add new capability pertaining to template library. Sp3 - reassessed successfully MainEx - Successfully utilized operationally
EBO_1.26	SDT - Provides a capability to save the current window as an image.	Demonstrate the ability to save the current window as an image		S	Sp3 - Saved the file as a JPEG. This has the same functionality as Alt-PrnScr. MainEx - Not reassessed

#	Capability	MOP/MOE/THRESHOLD	Location	Eval	Comments
EBO_1.27	OAT - Provides a capability to view the probabilistic assessments of blue COA elements and their impact on red COGs.	Demonstrate the ability to view probability profiles for actions and causal links.		S	Sp3 - Analyzed blue interventions against red COG nodes. Adjusted blue intervention causal strength and probability profile was updated accordingly. MainEx - Although probability profiles reflected impacts of blue interventions and causal parameter modifications, 8AF noted potential deficiencies in the probability profile algorithms with respect to reconstitution time modeling. The initiative provider agreed that refinement of the algorithms was necessary. Recommend accuracy of probability profile algorithms be validated during formal testing
EBO_1.28	OAT - Provides a capability to view the impact of evidence on the probabilistic assessments of blue COA elements and their impact on red COGs.	Demonstrate the ability to enter evidence and view the resulting probability profiles for actions and causal links.		S	Sp3 - assessed successfully (see comments for 1.27) MainEx - assessed successfully (see comments for 1.27)
EBO_1.29	SDT provides the capability for users to define COA comparison criteria and relative weights, enter scores for each phase of each COA, and automatically computes weighted totals for each COA. SDT also provides the capability to export the COA comparison matrix to MS Excel.	Demonstrate the ability to invoke a COA Comparison table within the tool that is pre-populated with COAs and phases in columns and rows respectively. Demonstrate the ability to alter the criteria used in the leaf-level cells. Demonstrate the ability to enter different weights for each criterion and each phase. Demonstrate the ability to enter values in the leaf-level cells, immediately triggering recomputation of the totals in the lower-most rows. Demonstrate the ability to export the COA comparison matrix headers and values to a comma-separated value file that may be imported into MS Excel.		S	MainEx (New Capability)- Scores were entered for multiple phases for the following criteria: Suitability, Feasibility, Acceptability, Completeness and Probability of Success. The Matrix was then successfully exported to MS Excel for operator comparison with other COAs
EBO_2.01	Provides a capability to enter queries and import data from the MIDB	Demonstrate the ability to enter queries and import information from the MIDB.		S	Sp2 - Multiple queries demonstrated. Sp3 - reassessed successfully MainEx - Successfully utilized operationally
EBO_2.02	Provides a capability to import COA and mission analysis data from the TBMCS/Strategy Management Services	Demonstrate the ability to import COA and mission analysis data from the TBMCS/SMS		S	Sp2 - Field inconsistencies with SDT and SPT Sp3 - The majority of SDT/SPT field inconsistencies have been resolved. However, some field implementations differ. Created a plan for export to verify import of the same plan. The exported data would not import. However, other previously created COA and mission analysis data imported successfully. MainEx - COA and mission analysis data successfully imported from the TBMCS/SMS
EBO_2.03	Provides a capability to export COA and mission analysis data to the TBMCS/SMS and import COA and mission analysis data from SMS.	Demonstrate the ability to export COA and mission analysis data to the TBMCS/SMS		S	Sp2 - Field inconsistencies with SDT and SPT Sp3 - The majority of SDT/SPT field inconsistencies have been resolved. However, some field implementations differ. MainEx - COA and mission analysis data successfully exported to TBMCS/SMS
EBO_2.04	Provides a capability to export candidate target lists to JTT	Demonstrate the ability to export candidate target lists to JTT		ND	Sp3 - Provider received JTT schema just prior to Sp3. Therefore, this capabiltiy is at risk for integration prior to MainEx. MainEx - Not integrated. However, this was not a required capability for 8AF operational use during JEFX04

#	Capability	MOP/MOE/THRESHOLD	Location	Eval	Comments
EBO_2.05	Provides the capability to display and export plans containing MOEs and indicators for interoperability with ISR Collection Management tools.	Demonstrate the ability to display and export plans in XML format with indicator and MOE information for use by ISR collection managers.		S	Sp2 - SDT currently supports exporting indicators in XML. Manual process will be used to input info into the PBA toolset in both Spiral 3 and MainEx MainEx - SDT successfully exported indicators in XML. However, in the absence of a machine-to-machine interface via SMS, a manual process was successfully used to input the information into the PBA toolset (AF2TK)
EBO_2.06	CAT - Provides a capability to revise the probabilstic assessment of the COA and its impact on red COGs based on evidence from ISR reports	Demonstrate the ability to input evidence from PBA/AF2TK via manual entry. Demonstrate the ability to view revised probabilities for COA objectives and red COGs.		P	Sp3 - AF2TK evidence received via email, manually entered and revised probabilities verified successfully MainEx - Additional assessment revealed that the probablistic assessment algorithm requires refinement to increase accuracy.
EBO_2.07	Provides the capability to import A2IPB products (NAIs, TAIs and Red COAs) for mission analysis visualization.	Verify A2IPB data can be displayed via map and table format.		S	Sp3 - A2IPB products were sent in XML via email. Once the email was received, the operator selected the XML and saved it in the directory. A2IPB products were then successfully displayed on the map and in table format. MainEx - Successfully utilized operationally
EBO_2.22	SDT provides the collaborative capability to upload and download plans to and from a relational database.	Demonstrate the ability to upload a new plan to the database. Demonstrate the ability to upload an existing file-based plan to the database. Demonstrate the ability to download a database plan.		S	MainEx (New Capability)- In a colaborative environment, a new plan was uploaded to the database while multiple users were waiting and verifying it. Existing file based plans were also uploaded by different users. Also the plans were downloaded by viarious users.
EBO_3.01	Provide the capability to build effects-based COAs	Demonstrate the ability to develop a COA that links tasks to desired effects.		S	See 1.01 through 1.10 Sp2: See 1.01 through 1.10 Sp3 - not reassessed MainEx - Successfully utilized operationally
EBO_3.02	Provide the capability to interface with other Air Force tools like IWPC through TBMCS	Demonstrate the ability to export a COA to TBMCS. Demonstrate the ability to load exported COA into another tool such as IWPC. Demonstrate the ability to import COAs from TBMCS that originated from another tool such as IWPC.		S	Sp3 - This may be a process issue. COAs can be exported to SMS successfully. However, import of strategy by other tools is TBD. MainEx - COAs were exported to SMS and successfully imported by IWPC
EBO_3.03	Provide the capability to import targets from MIDB and provide traceability of direct targets to affected targets.	Demonstrate the ability to import target information from the MIDB. Demonstrate the ability to associate targets with effects. Demonstrate ability to define direct target sets for tasks.		S	Assessed successfully, but not utilized operationally
EBO_3.04	SDT provides the collaborative capability for multiple users to edit the same collaborative plan from different workstations, with locks shown for in-progress edits, and real-time updates of changes pushed to all users` displays.	Verify that a user is able to lock and edit individual tabbed panes for Mission Analysis data (Specified Tasks, Implied Tasks, Resources, ROEs, CCIRs, Mission Statement, etc). When one user adds, deletes or edits a field in one tab, that tab shows up locked on another user`s workstation MA window, and vice-versa. When the user clicks OK to accent the changes, the other user`s tab is immediately unlocked and updated. Verify that a user is able to lock and edit individual plan nodes and collaboratively add, delete, paste nodes and update node properties.When one user adds, deletes, or pastes a node, the change should immediately show up on the other user`s workstation plan view. When one user starts typing edits to a plan node in the Properties editor, the node should show up locked in the other user`s view. When the first user applies the edits, the other user`s view should be immediately unlocked and updated.		S	MainEx (New Capability) - Users were able to lock and edit data panes. Indicators appeared on the other users workstations noting that some one has locked the data for editing. This allows the group to work together without interferring with each others work. Upon unlocking the other users were able to see the updated information. Nodes were also updated by locking out other users during the change and then presenting the new data once released by the orgiinal editor.
EBO_4.01	Provide the capability to import Joint strategy and mission analysis data and elaborate the Air Force component of the COA	Demonstrate the ability to import information from a Joint plan stored in TBMCS SMS.		S	MainEx - Successfully utilized operationally.

#	Capability	MOP/MOE/THRESHOLD	Location	Eval	Comments
EBO_4.02	Provide the capability to export an elaborated Air Force component COA to the Joint community	Demonstrate the ability to export information to a Joint plan in TBMCS SMS.		S	MainEx - Successfully utilized operationally.
EBO_4.03	Provide the capability to import Joint targets for inclusion in the elaborated COA	Demonstrate the ability to import target information from a joint target database such as the MIDB.		S	MainEx - Successfully utilized operationally.